Management of the Internet and Complex Services

European Sixth Framework Network of Excellence FP6-2004-IST-026854-NoE

Deliverable 8.5
Selected Economic Management Mechanisms

The EMANICS Consortium

Caisse des Dépôts et Consignations, CDC, France
Institut National de Recherche en Informatique et Automatique, INRIA, France
University of Twente, UT, The Netherlands
Imperial College, IC, UK
Jacobs University Bremen, JUB, Germany
KTH Royal Institute of Technology, KTH, Sweden
Oslo University College, HIO, Norway
Universidat Politecnica de Catalunya, UPC, Spain
University of Federal Armed Forces Munich, CETIM/UniBwM, Germany
Poznan Supercomputing and Networking Center, PSNC, Poland
University of Zürich, UniZH, Switzerland
Ludwig-Maximilian University Munich, LMU, Germany
University College London, UCL, UK
University of Pitesti, UPI, Romania

For more information on this document or the EMANICS project, please contact:

Dr. Olivier Festor
Technopole de Nancy-Brabois — Campus scientifique
615, rue de Jardin Botanique — B.P. 101
F—54600 Villers Les Nancy Cedex
France

Phone: +33 383 59 30 66
Fax: +33 383 41 30 79
E-mail: <olivier.festor@loria.fr>

Document Control

Title: Selected Economic Management Mechanisms

Type: Public

Editors: Martin Waldburger, Burkhard Stiller

E-mail: waldburger@ifi.uzh.ch, stiller@ifi.uzh.ch

Authors: Marinos Charalambides, Javier Rubio Loyola, Guilherme Sperb Machado, Helmut Reiser, Thomas Schaaf, Joan Serrat, Burkhard Stiller, Martin Waldburger (alphabetic order)

Doc ID: D8.5-v1.0

AMENDMENT HISTORY

Version	Date	Author	Description/Comments
V0.1	18.03.2009	Martin Waldburger	Document Structure, initial draft content for SLAP'N, GridAcc, IELOS
V0.2	16.04.2009	Helmut Reiser, Martin Waldburger	GridAcc accounting database attributes, IELOS key assumptions
V0.3	05.07.2009	Thomas Schaaf, Martin Wald-burger	SLAP'N ITSM extension, GridAcc netflow approach
V0.4	22.07.2009	Martin Waldburger, Burkhard Stiller	Updated document structure, integrated content for GridAcc, IELOS, SLAP'N, initial content in introduction, global outcome
V0.5	21.09.2009	Marinos Charalambides, Joan Serrat, Javier Rubio Loyola	IELOS structure and content update, PRIPOL structure and initial content
V0.6	30.10.2009	Martin Waldburger, Guilherme Sperb Machado, Thomas Schaaf, Helmut Reiser	Major GridAcc and SLAP'N content update
V0.7	12.11.2009	Joan Serrat, Javier Rubio Loyola, Martin Waldburger	Full integrated PRIPOL section
V0.8	03.12.2009	Marinos Charalambides, Martin Waldburger	Full integrated IELOS section
V0.9	27.12.2009	Martin Waldburger, Guilherme Sperb Machado, Thomas Shaaf	Full integrated SLAP'N section, full integrated GridAcc section
V1.0	02.01.2010	Martin Waldburger	Introduction, summary, final editing

Table of Content

1 Introduction

The document at hand is the final deliverable in a row of altogether five deliverables within the frame of EMANICS WP8, all of which concerned with various stages in an overall effort to develop a comprehensive model for economic management mechanisms. The first two deliverables were focusing on building a solid basis for the design of this model—on the one hand by giving an elaborate overview of existing key management models and mechanisms (D8.1 [26]), on the other hand by achievement of steps and actions undertaken to integrate economic and technical management mechanisms in a homogeneously interacting approach (D8.2 [25]).

D8.3 [30] and D8.4 [31] adopted this basis and delivered, in a first step, the draft extend model for economic management including important enhancements going beyond the traditional models of mechanisms and functionality, followed by the blueprint and evaluation of the final model and those economic management principles determined. This included significant contributions in selected, nonetheless integrated, work efforts in an overall economic management model by means of a strong emphasis on challenges which became apparent and needed in a commercial context. Issues of auditing, pricing, and cost accounting constitute core management functions in this overall economic model which, so far, typically remained partly neglected or fully abstracted by established network management model like the FCAPS (Fault, Configuration, Accounting, Performance, and Security) model.

Consequently, D8.5 carries on the highly successful tradition of those previous achievements made and the overall model developed by addressing selected economic management mechanisms of particular importance as concluded in D8.4. The accordingly covered four projects of D8.5—GridAcc, PRIPOL, IELOS, and SLAP'N—perfectly match the economic management model as well as the overall WP8 objectives. Thus, these efforts are in-line with a common target to extend the inter-operation model considering mechanisms for accounting, charging, Service Level Management (SLM), and Internet services—a model which aims to provide a customer service management architecture serving as the integration platform for a production environment, and a model that achieves a mapping approach for QoS parameters between and across organizational boundaries and platforms for the provision of guaranteed QoS at agreed cost and as captured within the appropriate set of Service Level Agreements (SLA).

In particular, D8.5 documents results obtained within the frame of the two concrete tasks attributed to WP8 efforts in 2009 as listed and defined subsequently:

- *Task 8.6: Functional Evaluations* Investigation of the functional strength of available mechanisms (*e.g.*, admission control, accounting, auditing, SLA management, performance management). Rating of charging and pricing approaches for new services.
- *Task 8.7: Refinements of Selected Mechanisms* Investigation on the selected integration mechanisms within economic approaches with the existing and newly developed technical mechanisms known from network management.

Table 1 presents the respective overview of those four projects covered in WP8 in 2009 and lists for each the set of participating partners and assigned tasks Task 8.6 and/or Task 8.7. All projects address Task 8.6, thus, they are concerned with an in-depth investigation of an economic management mechanism each: GridAcc emphasizes on selected improvements in the area of cost accounting for Grid systems. PRIPOL and IELOS address business-driven management by policies and pricing mechanisms, while IELOS and SLAP'N cover

Table 1: WP8 Projects Covered with Partners and Tasks Assigned

Project	Participants	T8.6	T8.7
GridAcc	UniZH, LMU/LRZ	X	X
PRIPOL	UPC, UniZH	X	X
IELOS	UCL, UniZH, UPC	X	
SLAP'N	LMU, UniZH	X	

the management of international service contract conclusion. GridAcc and PRIPOL follow up on similar projects of earlier phases in WP8 so that these projects address Task 8.7 in addition to Task 8.6. Accordingly, the remainder of this deliverable is structured as follows: Section 2 to Section 5 document the GridAcc, PRIPOL, IELOS, and SLAP'N projects each, while Section 6 summarizes and concludes the respective obtained results.

2 GridAcc — Grid Accounting

Large-scale service-oriented computing in a local computing center context of a computational Grid or across multiple domains within dynamic VOs (Virtual Organization) clearly states the need for the suitable economic management mechanisms to be in place. Accounting of services provided and resources used constitutes a management mechanism of key importance. This is due to the fact that accounting records form the basis for further related, time- and content-wise dependent management tasks. In particular, this includes charging and billing as well as analysis and optimization.

Accordingly, a first phase in GridAcc embraced the design of a highly flexible accounting model [11] as follows: As the initial starting point, existing accounting approaches applicable to the specific needs of computing centers and Grid systems were analyzed and compared. This comprehensive related work study has revealed that issues of technical precision and project-driven optimizations had been focused so far, while neither approach showed sufficient support of multi-domain or virtualization concepts. In addition, neither approach funded on the suitable accounting principles from business domains. Consequently, these identified gaps led to the development of an accounting model suitable for computing centers and dynamic VOs. This model combines principles of technical and business accounting by means of Activity-based Costing (ABC) service constituent parts and defined accountable units which take characteristics of a considered technical resource into account.

Driven by a successful preliminary functional evaluation of the developed accounting model, a full-fledged model application and evaluation case was conducted in the second phase of GridAcc. To this aim, the model was applied to the context of the Leibniz Supercomputing Centre (LRZ) in Garching, Germany, where the former lead was a member of LMU (Ludwig Maximilians University of Munich LMU, Germany). Accordingly, the applicable in-depth methodology to apply and evaluate the accounting model was determined, then the model was applied to the LRZ whereby gained results were fully documented by means of an elaborate full cost calculation, and finally those results gained were discussed and evaluated according to those previously defined evaluation criteria in the areas of model functionality, model parametrization, and model application context. The work—which led to the paper "Evaluation of an Accounting Model for Dynamic Virtual Organizations" published in the Journal of Grid Computing [28]—was concluded to be highly successful: The model was found to constitute a valuable means to (a) reliably determine full costs of a considered service with a reasonable amount of required input data, and to (b) determine organization-internal optimization potential. Consequently, GridAcc helps operators of computing centers to identify costs incurred by a specific service provided and to identify potential inefficiencies in their cost structure. Therefore, GridAcc may serve as a means to increase competitiveness.

Besides these positive overall conclusions drawn, the application case revealed a number of dimensions for further work:

- Extension of the model in order to consider load balancing aspects.
- Extension of the concept of quality premiums to better support competition for resources.
- Consideration of costs caused by unused but not attributable resources in a more fine-granular way.
- Definition and integration of generally applicable set of metering points for technical accounting.

These areas mentioned constitute the starting point for the ongoing third phase in GridAcc in order to further improve the developed and successfully applied accounting model for computing centers and Grid systems. Consequently, GridAcc is focusing in 2009 on **cost impacts through inter-job effects and resource scarcity**. Section 2.1 explains in further detail what such cost impacts might comprise. This is followed by a consideration of the available accounting-relevant data at the LRZ (*cf*. Section 2.2), leading to a detailed scope definition in Section 2.3. The accordingly determined system design and the respectively obtained achievements are described in Section 2.4 and Section 2.5, respectively.

2.1 Cost Impacts Through Inter-job Effects and Resource Scarcity

The main purpose of this scenario is not to form the scenario to be used in a future GridAcc paper, but to analyze - from a cost accounting point of view - the overall impact of inter-job effects and resulting resource scarcity on cost. Most input information used to sketch this scenario was taken or derived from the LRZ webpage documentation, in particular:

- HLRBII-related information on hardware [18]: Especially important are number of nodes in phase 2 (9728 nodes) and number of compute partitions (19).
- Usage statistics according to job classes, CPUh, and system time used [19]: All further data calculations in relation to usage statistics are depending on February 2009 data. Thus, the assumption of February 2009 as a month with representative data was implicitly made.
- Job configuration parameters of relevance for this scenario [17]: Especially of importance here are available job classes and upper runtime limits (cf. Table 1), partition characteristics (1 interactive partition, 18 batch partitions) as well as ncpus (number of CPU cores) and walltime (needs to be within maximum runtime limit) parameter.

2.1.1 Job Classes and Upper Runtime Limits

A basic assumption is taken for the scenario as follows: For reasons of simplicity (with respect to a smoother "fit" into partitions) and driven by that information gained from partitions and usage statistics (*cf*. respective subsections further on), these job classes introduced are assumed to use the following static amount of cores:

- N64: 63 cores
- N128: 127 cores
- N256: 254 cores
- N510: 508 cores
- N1020: 1016 cores

2.1.2 Compute Partitions

There are 19 partitions in total, thus each embracing 512 cores on average. Not all cores are available for computing, however. There is a certain overhead to be considered. Furthermore, two types of partitions are differentiated, interactive partitions and batch partitions.

Only one partition is of type interactive partition. Within that partition, there are 32 login cores assigned (nodes which are heavily loaded). These cores are not foreseen to be used for actual computing tasks, but for compiling and testing. The actual computing within this partition is done on the (remaining) 476 cores. These are dynamically assigned in either a so-called shared or dedicated mode by a scheduler—the Altair PBS Pro. There is no

Table 2: Run Time Limits and Default Partitions for Jobs [17]

Job Class	Job Size	Runtime limit (hours)	Default Partition	Runtime limit (hours)
N64	1-64	48	Density or Bandwidth	Serial or small-scale parallel runs. May also be scheduled to fill small holes in partitions. Use multiples of 2 (even for serial runs!), and if you specify -l density=true, use multiples of 4.
N128	65-128	48	Density or Bandwidth	Small-scale parallel runs. Will be run on a single partition.
N256	129-256	48	Density or Bandwidth	Medium-scale parallel runs. Will be run on a single partition.
N510	257-510 (257-508)	48	Bandwidth	Medium-scale parallel runs; can be run on a single partition if properly configured (ncpus=...). If you specify -l density=true, only up to 508 cores can be requested.
N1020	511-1020	32	Bandwidth	Large-scale parallel multi-partition runs. (-l density=true is possible, but not recommended).
N2040	1021-2040	12	Bandwidth	Large-scale parallel multi-partition runs (-l density=true is possible, but not recommended).
N4080	2041-4080	12	Bandwidth	Very large-scale parallel multi-partition runs. (-l density=true is possible, but not recommended).
special	>4080	no explicit limit	Bandwidth	This job class is for jobs which need more than 4080 CPUs or more wall time than the job classes specified above. You need to apply for access for this queue. This type of job may be scheduled to a mixture of high-density and high-bandwidth blades, hence special care is required in setting this kind of job up properly.
np_S64	1-64	8	Density	Small-scale runs on the node which is used for interactive jobs at day. These jobs will only run at night (8 pm - 8 am). It is necessary to specify the nightrun property to differentiate the job from a standard small-scale parallel job.
N64L	1-64	96	Density	Serial or small-scale parallel runs with wall clock requirements of more than 48 hours. Use multiples of 4 (even for serial runs!).

detailed information available based on what criteria the scheduling algorithm works. Only the following statement can be found on the LRZ webpage [17]: "LRZ will favor large parallel jobs with respect to scheduling."

In addition to the single interactive partition, there are 18 batch partitions. Per partition, two cores are reserved for the OS, leaving 510 cores available in case of so-called high-bandwidth blades are used or leaving 508 cores available in case of so-called high-density blades are used, respectively. Five partitions use high-density blades, while 13 partitions use high-bandwidth blades.

Assumptions taken for the scenario:

- For reasons of simplicity, the scenario does not differentiate:
  - → partition types (meaning all partitions are assumed to be the same type, namely batch partitions),
  - → blade types (meaning all partitions are assumed to feature the same amount of cores, namely 508 cores).
- For reasons of simplicity, the scenario will presume the existence of four partitions only (instead of 19), so that inter-job effects become visible faster. The total number of cores, thus, is reduced in the scenario to 4 x 508 cores = 2032 cores.
- Since no scheduling algorithm is available, a simple (probably not the most efficient though) algorithm for scheduling jobs is assumed (see Figure 1).

```
The algorithm for placing newly requested jobs or jobs from the backlog (the queue) is
triggered equally by two events:
   A new job request is received by the scheduler.
   A running job was terminated. In fact, not the actual, terminated job is detected and
   will trigger the event, but rather the time when a reservation of a running job has
   finished.

Once the trigger is released:
Label: 1
If there is a job waiting in the backlog:
   If trigger is for new job:
            Do not assign job, put job to backlog:
   Take the oldest job in the backlog.
   If this job is of class N1020:
            If two full partitions (priority: left-to-right) are free at this moment:
                        Assign job, remove from backlog, Goto 1.
            Else:
                        Do not assign job, leave in backlog, break.
   Else:
            If there is space according to the job ncpus value free at the moment (condition:
            job to be placed within a partition, not overlapping two partitions; priority:
            left-to-right):
                        Assign job, remove from backlog, Goto 1.
            Else:
                        Do not assign job, leave in backlog, break.
Else (backlog empty):
   If trigger is for new job:
            If this job is of class N1020:
                        If two full partitions (priority: left-to-right) are free at this
                        moment:
                                    Assign job, break.
                        Else:
                                    Do not assign job, put job to backlog, break.
            Else:
                        If there is space according to the job ncpus value free at the moment
                        (condition: job to be placed within a partition, not overlapping two
                        partitions; priority: left-to-right):
                                    Assign job, break.
                        Else:
                                    Do not assign job, put job to backlog, break.
   Else:
            Break (nothing to do).
```

Figure 1: Simple Example of a Scheduling Algorithm

2.1.3 Job Classes

Figure 2 presents the primary input data (as published by the LRZ for February 2009 [19]) to calculate job type occurence, job length (walltime) and to estimate number of CPU cores per job type. February 2009 has 28 days, which translates to 672 h of possible compute time.

HLRB (hlrb+hlrb2i) Nutzung für den Zeitraum von 2009-02-01 bis 2009-03-01

	Jobs		Systemzeit	
Jobklasse	Anzahl	%	CPU-Std.	%
N1020	242	2.74	1499127.24	27.71
N256	683	7.74	1388823.46	25.67
N64	5388	61.03	1183062.55	21.87
N128	1632	18.48	728366.61	13.46
N510	179	2.03	349350.79	6.46
N64L	207	2.34	135603.55	2.51
N4080	12	0.14	59621.29	1.10
N2040	24	0.27	46471.54	0.86
I	460	5.21	15955.35	0.29
special	1	0.01	3168.92	0.06
t_eval	1	0.01	0.00	0.00
Summe	8829	100.00	5409551.30	100.00

Figure 2: HLRB II Usage Statistics Overview for February 2009 [19]

Assumptions taken for the scenario:

- For reasons of simplicity, only the five most important (criteria: top-five in relative system time used) job classes are considered for the scenario. These five classes represent more than 95% of system time used in February 2009 (accumulated). These five classes are:
  - → N1020
  - → N510
  - → N256
  - → N128
  - → N64
- Based on the information in the above usage statistics, the following was calculated for the five considered job classes (assumptions mentioned for first class, the same hold for all classes):
  - → N1020: 1499127 CPUh on 242 jobs --> avg. 6195 CPUh/job (assuming even distribution), avg. 6.1 h/job, assuming ncpus = 1018
    - Assumed wallclock parameter (for reservation): 7.625 h (=6.1 x 1.25)
    - Every 2.78 h a new job of this class is requested (assuming new jobs entering at a constant, evenly distributed range).
  - → N510: 349351 CPUh on 179 jobs --> avg. 1952 CPUh/job, avg. 3.8 h/job, ncpus = 508
    - wallclock: 4.75 h
    - Every 3.75 new request.
  - → N256: 1388823 CPUh on 683 jobs --> avg. 2033 CPUh/job, avg. 8 h/job, ncpus = 254

- wallclock: 10 h
- Every 0.98 h new request.

→ N128: 728367 CPUh on 1632 jobs --> avg. 446 CPUh/job, avg. 3.5 h/job, ncpus = 127
- wallclock: 4.375 h
- Every 0.41 h new request.

→ N64: 1183063 CPUh on 5388 jobs --> avg. 220 CPUh/job, avg. 3.5 h/job, ncpus = 63
- wallclock: 4.375 h
- Every 0.12 h new request.

2.1.4 Scenario

Figure 3 shows the outcome of these assumptions taken and applied to the scenario with four partitions. It is important to note that Figure 3 is to be read in a bottom-to-top (dimension: time) and left-to-right (dimension: CPU cores, four partitions, each partition separated by a vertical black line). Horizontal lines represent those moments (and trigger events) when a new job request arrives. Coloring is attributed as follows:

- Dark green/brownish: Job class N64
- Blue: Job class N128
- Pink: Job class N256
- (Lighter) green: Job class N510
- Red: Job class N1020

Blocks use the same coloring scheme as the horizontal lines, while blocks represent attributed jobs. A job's horizontal dimension, accordingly, reflects its number of CPU cores occupied, its vertical dimensions what (walltime) was reserved for this job. Blocks are numbered. These numbers correspond to the order of request received as seen by the scheduler. When zooming into Figure 3 on the right lower corner, the first 40 request for new jobs are numbered. These 40 blocks are attributed as visualized. In the right lower corner, there are in addition requests marked by a star symbol whenever a newly requested job cannot be attributed and must be, thus, put to the backlog.

2.1.5 Input and Scenario Conclusions

Driven by the main purpose of this scenario, which is to show cost impact of inter-job effects and resource scarcity, the following conclusions may be drawn:

- Depending on the actual scheduling algorithm and based on those many assumptions taken, there are may be considerable inefficiencies to be observed. With such inefficiencies, the influence on costs become very well visible.
- In particular, long-lasting reservations and a strict timely order in the scheduling algorithm seem to have negative influence on costs.
- In order to proceed - which will lead to the actual scenario to be addressed - there would be specific, detailed knowledge needed about the LRZ scheduling algorithm. As this information is not made available by the software manufacturer behind PBS Pro, an investigation into scheduling algorithms is needed, based on what an algorithm has to be chosen.

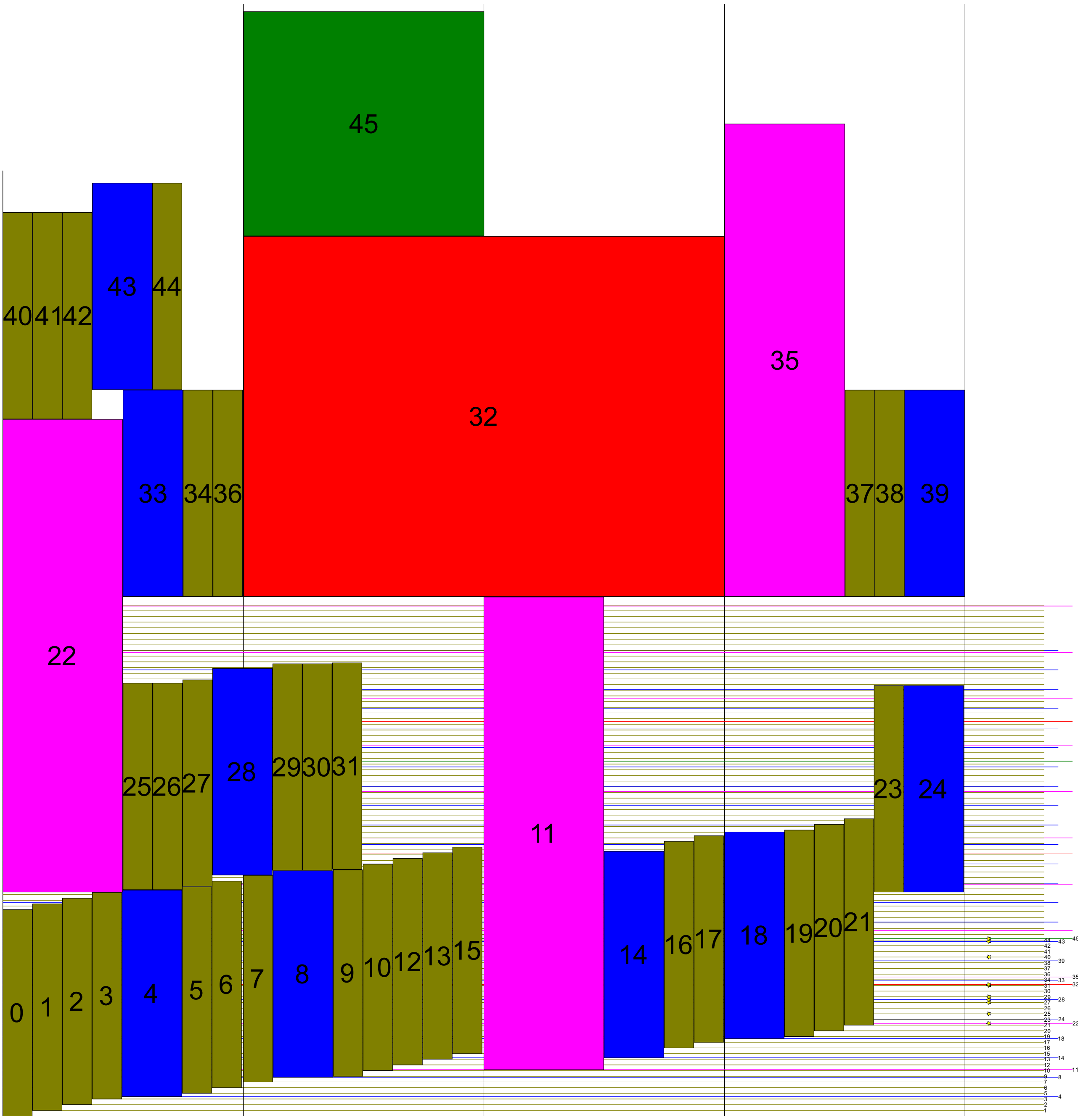


Figure 3: Scenario Showing Inter-job Effects and Resource Scarcity Based on Explained

2.2 Accounting-relevant Data Available at the LRZ

2.2.1 Accounting Database

The LRZ uses two accounting-oriented databases of relevance here, the HLRBII accounting database and the PBS Pro accounting database. Data retrieval and management of the latter is reported to be less comfortable than handling data from the first. Effort might possibly be a reason for the very existence of the HLRBII accounting database, while this thought represents speculation at this point in time only. In the following, the HLRBII accounting database is considered exclusively. Therefore, whenever the term "accounting database" or "database" is referred subsequently, this reference implies the HLRBII accounting database within the scope of this section.

The database is fed from the respectively appropriate information filtered from amounted raw data. Typically, raw data sums up to a total of 1.5 to 3 MBytes per day, with (statistical) outliers ranging up to a raw data amount produced of 6 MBytes a day. The following list documents a number of extracted database fields with their respective use and interpretation. Each entry in the database represents a single job.

Since the database is in use for a longer time, it needs to cope with multiple data sources. This implies that the database sees partially overlapping attributes or attributes that are maintained for historic reasons, which however are not used actively today. Attributes in the database reflect variables from SGE (previously in use) and from PBS Pro (currently in use).

- bd_number: Main identification number for relevant jobs. If set to value of "3" then a job was run on PBS Pro/HLRBII. This work's focus is set on HLRBII so that relevant jobs are filtered by a lookup of jobs where bd_number equals 3.
- hostname: String to identify the partition a job is run on. In case of jobs spanning multiple partitions, hostname indicates the first partition used. There is, however, no guarantee that this information is made available in the database for every job.
- os_group, owner: Main database fields to identify a system user (owner) related to a job as well as to identify the user group (os_group) the respective system user belongs. These values represent standard UNIX system mechanisms. User groups are typically formed for a project.
- start_time, start_time_str, end_time, end_time_str, submission_time, submission_time_str: Log the respective moments in time when a job was submitted initially (submission_time), when it was started (start_time), and when it ended (end_time). Time is represented in UNIX format or, in case of database attributes ending with "_str", as a string in the format of YYYYMMDDHHMMSS.
- queue_time, queue_time_str, equeue_time, equeue_time_str: Logs the time (in seconds as in UNIX time format, as a string value in case of queue_time_str) when as job was queued into its current queue. The "e" in equeue_time/equeue_time_str stands for execution queue. Accordingly, equeue_time/equeue_time_str logs the time (in seconds as in UNIX time format and as a string, respectively) when a job became eligible to run, i.e. it was enqueued in an execution queue and was in the "Q" state.
- ru_wallclock: Logs the effective (as opposed to reserved) time a job used, measured in seconds.
- slots, ru_cores: The number of requested CPU cores is reflected by ru_cores, the number of CPU cores actually attributed by PBS Pro (the scheduler) is reflected by slots. Slots are attributed on the basis of vnodes, whereas a vnode may either comprise 2 or 4 CPU cores. Typically ru_cores is smaller than or equals the number of slots. If a job, however, requests only 1 CPU, or if a job requests more memory than is available by the requested CPUs, then PBS Pro attributes more slots than requested CPU cores. Accounting- and charging-relevant are slots, not ru_cores.
- cpu: Represents the total amount of CPU time used by the job for all processes on all vnodes (see slots/ru_cores for an explanation on what a vnode is).
- mem, maxvmem: mem represents the memory used (measured in bytes) by a job. The value of mem embraces the accounted memory map of the involved system process(es) including all libraries. maxmem represents the maximum, aggregate

amount of virtual memory (in bytes) used by all concurrent processes in a job. Documentation on maxmem is scarce, so that maxmem specifics remain unclear. maxmem is assumed, however, to reflect the accounted peak value of memory usage.

- accountable: Binary value to identify whether a job is accounted or not. A value of 1 implies that a job is accountable, a value other than 1 implies a job is not accountable. Accountable in this context means that a job may be charged, thus, non-accountable jobs are those jobs that may not be charged. Non-accountable jobs are typically jobs that saw some technical issues during execution, like partition failure, system crash - in general, issues not caused by a job's user.
- job_name, job_number: Attributes to identify a job by means of its attributed system user. job_name is a string value, job_number is a system user ID. In an array job, all jobs receive the same ID.
- qname: There are multiple queues in use in order to logically separate larger contexts. For instance, the D-Grid project has its own queue. Each queue is given a name, qname, by which a queue is identified.
- priority, granted_pe: These attributes reflect SGE variables and are not in use anymore.
- failed, exit_status: This attributes determine whether a job ran through or failed. An exit status of 0 signals normal execution. A value different from 0 indicates a failed job.
- category: category represents user requirements for a resource or resource default configurations calculated by a script (qsub), respectively. By means of these resource requirements the number of requested CPUs per chunk can be calculated (see [17] for information on submitting jobs including parameter settings such as select=<number of chunks>:<ncpus>). The actually attributed and accounted amount of CPUs is found in slots, however. More cores than requested are attributed if requested memory is greater than available memory on requested CPUs (each core has a maximum of 3896 MBytes of memory available).
- project: This is used for ID management purposes at the LRZ. Each project needs approval by the LRZ steering committee. A project receives an ID, which typically is reflected by a UNIX group ID (see os_group). Under that project/group ID, multiple user IDs are assigned for sub-projects.

2.2.2 Netflow-based Accounting

In addition to the database, netflow data was considered to be used for the purposes of this work, in particular with regard to network traffic caused by a job. The idea was to identify hosts by means of their (fixed) IP address and to collect netflow data for this set of identified IP addresses. For the combination of four main arguments, netflow as a possible accounting-relevant data source is not followed any further. None of these arguments would question netflow *per se*, whereas these arguments question its overall benefits in relation to its efforts/costs:

- As a typical job involves multiple nodes, a high number of flows (determining communication between a single source and destination IP) is to be expected. In a worst case scenario, where every node would communicate with each other (both directions, non-variable ports) and under the assumptions that this (internal) traffic is cost-relevant (e.g., with respect to maintenance labor) and that it can be monitored, a maximum of N*(N-1)*2 flows had to be accounted, N being the number of nodes/IPs. This number reflects a full mesh communication which is doubled by flows in both directions. These single flows would need to be mapped to a job. In case of larger jobs, the resulting effort is expected to be considerable.

- Experience at the LRZ shows that only a small part of all network communications is external, meaning flows are primarily within the nodes assigned to a job. Only if internal traffic is seen highly relevant to cost accounting, its benefit outweighs its effort. Furthermore, netflow data would typically cover flows involving IPs from outside, so that the majority of flows would not be visible to netflow by default.
- As there is a batch system in place, code can be executed only after a job becomes active. The time this happens is transparent to a user, first since PBS Pro deals with scheduling transparently, and second since the time span a requested job waits in a queue depends on other, external factors like the number of other job requests waiting. Job-relevant data, on the other hand, is typically copied by means of SCP to the SAN before a job request is submitted. This means that the cost-relevant activity of copying (typically large amounts of data) to the SAN is time-wise decoupled from job execution so that any neflow-based monitoring of that activity cannot be triggered by job submission or job execution events.
- Batch jobs profit from those means deployed at the LRZ to enable parallel execution. Even though a batch job runs in a dedicated environment, i.e., assigned cores are exclusively available to a single user for a limited time period, a netflow-based accounting could not separate traffic caused by different processes/threads running in parallel within that dedicated environment. It could only differ traffic caused by the overall dedicated resources (each node represented by an IP). Only if such a higher level accounting granularity is sufficient, netflow can be used.

2.2.3 Accounting Data Conclusions

While it was concluded that a netflow-based accounting of the traffic related to a considered job is not further followed for a number of reasons that question such an approach with respect to a beneficial cost value ratio, the accounting database deployed at the LRZ was found a valuable data source for the purposes of this work. The database is expressive enough to cover the relevant set of organizational (e.g., hierarchy of projects/groups, sub-projects/users) and technical (e.g., time periods, resources used) attributes needed for a proper cost accounting.

2.3 Modeling of New Assumptions

Based on the collected and discussed information on basic input (parameters like scheduling algorithm and job classes with arrival rates), scenario, and accounting database attributes, the scope of GridAcc is determined as visualized in Figure 4.

GridAcc, thus, is foreseen to take three main input sources:

- Scheduling algorithm: The scheduling algorithm in place shows significant impact on job-related costs. Depending on a specific algorithm's characteristics, inter-job effects and resource scarcity may be positively or negatively influenced. GridAcc needs to assume at least one scheduling algorithm, while support for exchangeable, multiple algorithms is preferable.
- Resource parameters: Not only the scheduling algorithm, but also those many different input parameters as presented in Section 2.1.1 through Section 2.1.4 show a large impact on job costs. These parameters deal on the one hand with characteristics of the infrastructure available at the LRZ (e.g., partitions) and, on the other hand, with job specifics (e.g., job arrival rates). Key parameter settings need to be reflected in GridAcc, again, preferably in a way that allows parameters to be changed.

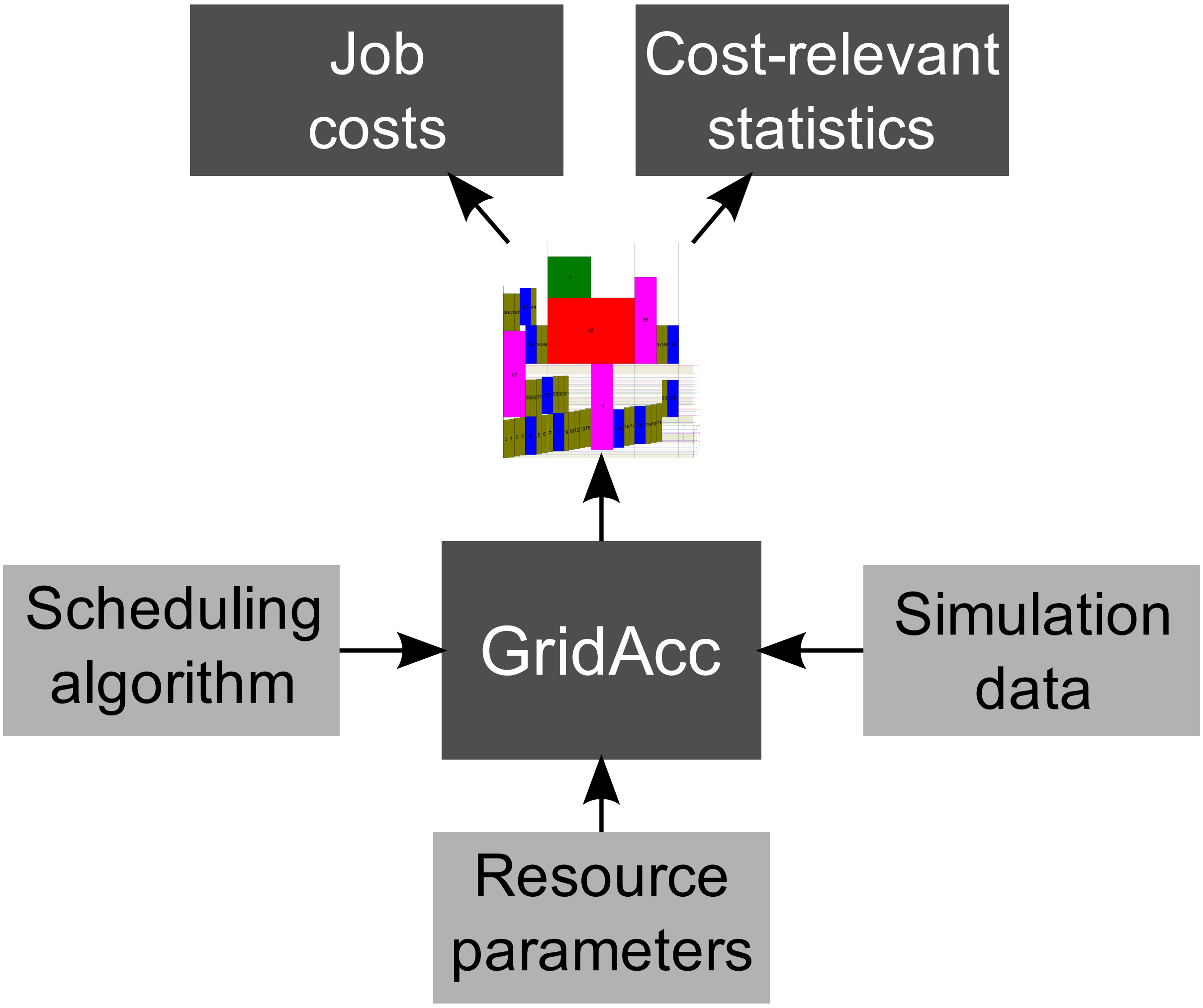


Figure 4: Foreseen Input and Output in GridAcc

- Simulation data: Scheduling algorithm and resource parameters need to be applied on a data basis in order to produce meaningful results. To this end, an accounting database snapshot is used as simulation data.

Driven by these three input sources, GridAcc is supposed to produce simulation results. In the first place, a graphical representation of the simulation is to be generated, similar to that (manually generated) visualization of Figure 3. In addition, the simulation shall determine these costs that are not directly attributed to a given job, but that are caused by inter-job effects and resource scarcity. Thus, the sum of direct and indirect job costs has to be determined for a considered job. Furthermore, the simulation shall allow for a number of cost-related (in a wider sense) values to be calculated. This includes, for instance, the average time a job of class X remains in a waiting queue or the average amount of unused CPUs.

2.4 System Design

In order to better understand how the three aforementioned input sources (scheduling algorithm, resource parameters, and simulated data) interact with the GridAcc system, this section presents the high-level GridAcc system design taking into consideration a general Grid environment (which can be, for example, the LRZ).

Figure 5 presents an overview of the system, showing how GridAcc is positioned in a Grid environment. Note that the arrows between entities represent the information exchange by a given interaction. For example, in the first step, a user may send some jobs to a Grid front-end in order to deploy these jobs in a large scale. The Grid front-end communicates with the scheduler that orchestrates job distribution for those Grid resources available. The scheduler is called “Actual Job Scheduler” since it is the existent component inside the Grid environment that can define, based on some parameters, where and when jobs will be executed among the available Grid resources. The word “actual” is merely a manner to not create any misunderstanding with “Scheduling Analyzer”, which will be described later on.

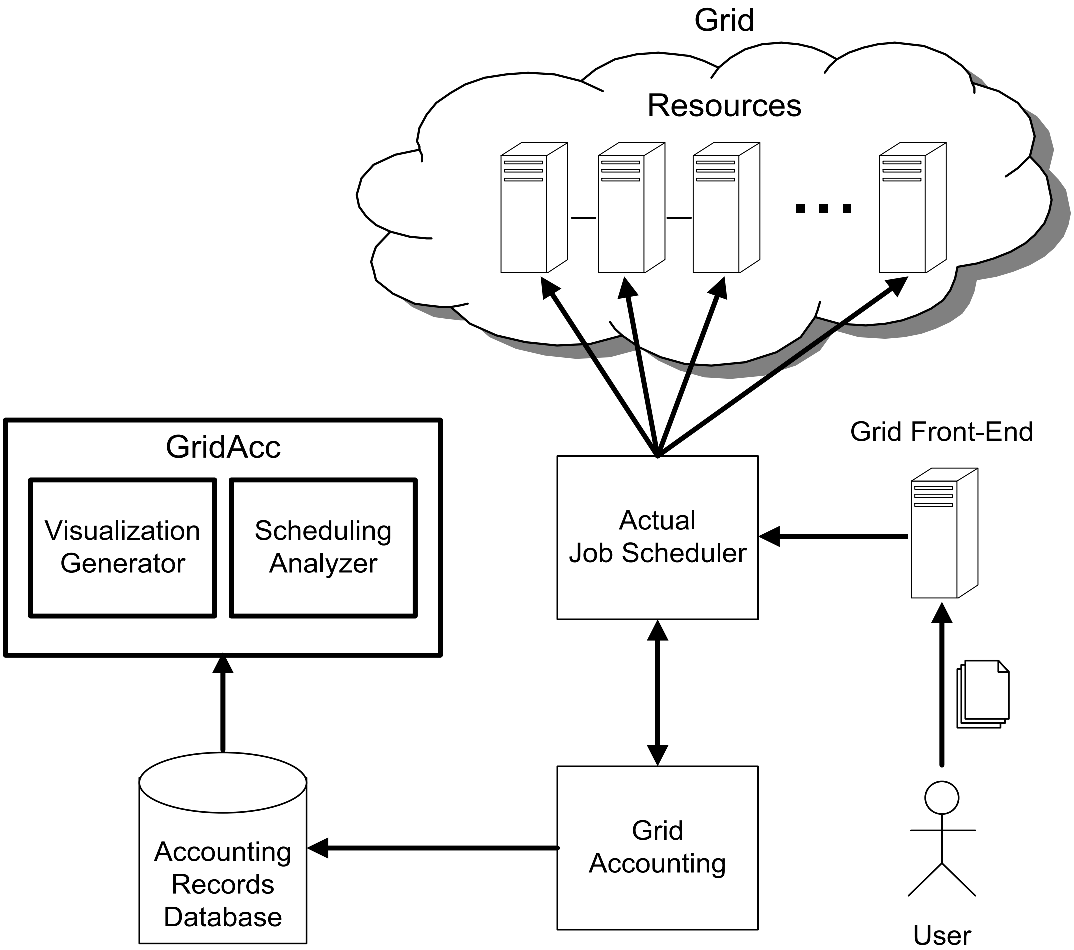


Figure 5: Overview of the GridAcc System in a General Grid Environment

While the “Actual Job Scheduler” distributes jobs, there is also the “Grid Accounting” that accounts how much/many Grid resources are being consumed by users. The accounting is made through a monitoring process, observing what was submitted to the scheduler, for example. All accounting records processed by the “Grid Accounting” component are persisted in a database called “Accounting Records Database”. Such database is of primary importance in the GridAcc context, and should be maintained in a secure, standardized, and reliable environment to avoid re-work or inconsistency.

What was described up to now may be faced as a general Grid environment as mentioned before, except for the lack of further details due to the sake of simplicity. However, one of the main goals for the system design is to build a component that supports, in terms of decision-making, a given actual Grid environment without significant technical interference.

Therefore, the GridAcc component is created aiming to analyze cost impacts through inter-job effects and resource scarcity in Grids interacting just with the “Accounting Records Database”, and not having any direct relation with the “Actual Job Scheduler”. Since GridAcc aims to be a decision-support component—meaning to help Grid operators understand what are the weak points on the on-production scheduling algorithm from a cost perspective—, it collects usage accounting records and then analyzes it based on some parameters which can be, for example, prioritization of time, better utilization of resources (in terms of having less idle CPUs), etc.

GridAcc has two sub components called “Scheduling Analyzer” and “Visualization Generator”. The “Scheduling Analyzer” shows higher relevance to generate final output since it may implement scheduling algorithms to analyze whether there is different ways to allocate resources compared to the way used by the “Actual Job Scheduler” component. The “Visualization Generator” takes such analysis and transforms it into a graph which facilitates visualization of, for example, inefficient CPU allocation in terms of CPU speed variation. Basically, the “Visualization Generator” has the responsibility to create an output that easily shows what is the differences related to what was generated by the analyzer compared to what was generated by the “Actual Job Scheduler”.

It is important to mention some aspects related to the “Scheduling Analyzer” component:

- It should be possible to integrate different scheduling algorithms to perform the analysis. For example, the analyzer can take accounting records from a month and analyze its cost impacts if applied with a Round-Robin algorithm giving priority to small jobs.
- The analyzer should be flexible in the sense that it can have resource parameters input on-demand. For example, the “job arrival rate” should be different to observe the impact caused when applied with an other scheduling algorithm.
- An interface between accounting records—in other words, the simulated data—and the “Scheduling Analyzer” should be well-defined (standardized).

2.4.1 Choosing Scheduling Approaches

There are many scheduling approaches found in computer science literature. For instance, [14] provides an embracing overview. Despite the wide range of explored scheduling approaches, there is no universal “best” scheduling algorithm, and many solutions use extended or combinations of the existent basic scheduling algorithms. Some examples of basic scheduling algorithms are:

- First In First Out: queues jobs in the order that they arrive in the “ready queue”.
- Shortest remaining time: the scheduler arranges jobs with the least estimated processing time remaining to be next in the queue. This requires advanced knowledge or estimations about the time required for a job to complete.
- Fixed priority pre-emptive scheduling: assigns a fixed priority rank to every job, and the scheduler arranges the jobs in the ready queue in the order of their priority. Lower priority jobs get interrupted by incoming higher priority jobs.
- Round-robin: the scheduler assigns a fixed time unit per job, and cycles through them.
- Multilevel Queue: used for situations in which jobs are easily classified into different groups. For example, two types of jobs may have different response-time requirements, and then they also may have different scheduling needs.

In order to extend or make combinations of these widely known scheduling algorithms, the primary criteria must be to obtain knowledge about what—according to what optimization dimension—to prioritize. This may vary for each specific job that is submitted to a Grid.

However, since using different scheduling algorithms at the same time is unrealistic and unfeasible in a Grid environment, criteria shall be established in order to observe and come up with the best approach for the whole system. Here, "criteria" is a term used envisaging to optimize job allocation to the Grid resources. To better understand it, a few example criteria are listed as follows:

- Homogeneous computing resources (probably it will generate less job errors, for example)
- Constant computing power (no processor speed variation when choosing multiple resources for one job)
- No inter-domain traffic
- Low latencies
- Less bandwidth consumption
- Less time span between start and end of a job
- The maximum of parallel jobs
- Less response time for a submitted job (allocating as many resources available to make the response time better)

Also, some criteria should be admitted when looking specifically at the scheduling algorithm in the "job choosing" phase. Examples of job prioritization include:

- Allocate big jobs or small jobs first
- First, allocate jobs that are not "night-run" jobs
- First, allocate jobs with less estimated time
- Always give more priority in the system for a certain job class (for the institute of physics, for example)

These two levels of criteria are different and influence job allocation. Thus, before choosing any approach, key system characteristics must be know, such as whether a system mainly receives many types of job classes, varying in multiple parameters like estimated execution time, job size, or restrictions on executing parallel jobs at the same resource.

2.5 Preliminary/Final Achievements and Next Steps

GridAcc has finalized its requirements, scenarios where the system is highly important, and the system design. Accordingly, the relevant set of next steps—some of which have already been started—has been determined as follows:

- Implement a prototype where both sub components of GridAcc produce the respective expected results. At the end, Grid operators should take advantage of the developed system to help on making decisions over the on-production scheduling algorithm. In other words, after further analysis using GridAcc an operator may decide to better adjust the actual job scheduler to achieve better results.
- Evaluate the solution using actual and on-production data. The evaluation should be done using a large amount of data since decision-support solutions are typically subject to interpretation: depending on the selected set of parameters to prioritize, and depending on the accounting data considered, analysis may be biased.
- Possible submission to the 9th Conference of Telecommunication, Media and Internet Techno-Economics (CTTE2010, conference website: http://www.ctte-conference.org) in order to publish achieved GridAcc results.

3 PRIPOL — Pricing by Policies

This section presents key results obtained within the frame of PRIPOL. PRIPOL addresses a business-driven approach to network management. The subsequent sections reflect an extract of the paper *"Business-driven Management of Differentiated Services"* [15] submitted to and accepted for publication at the 12th IEEE/IFIP Network Operations and Management Symposium (NOMS 2010).

3.1 A Business-driven Approach for Fixed Networks Management

Differentiated Services (DiffServ) have been proposed as a scalable approach for providing Quality of Service (QoS) in IP networks. Nevertheless, DiffServ only provides the instruments to set QoS attributes but not the control mechanisms to play with these QoS to properly manage the network. These control mechanism have to be developed. Otherwise, the absence of control based on, for example, profit, service admission, and resource limitation, may lead to excessive traffic injection that can result to resource starvation and consequently network congestion [6].

Network congestion prevention and solving, control of service subscriptions and invocations, and dynamic traffic engineering functions have been the centre of study in intra-domain [27] and inter-domain [13] management solutions. Although these and other solutions demonstrate how services can be delivered with QoS guarantees, the incremental efforts to elevate their business value have remained largely unexplored.

This work addresses the need to bridge the gap between business and configuration management. We describe an approach that identifies relevant business indicators that in turn are used in the process of designing configuration policies to regulate the admission of new service subscriptions and the traffic accepted from already active services.

The center of our approach consists of identifying a set of mapping functions between business indicators and parameters of configuration policies. In that way, a system administrator giving specific importance weight to the available business indicators is directly affecting the policies to be enforced to achieve such indicators. But before describing these mappings in detail we expose the mechanisms by means of which these policies will manage the network. Finally we demonstrate through simulation, the effectiveness and practicality of the approach.

3.1.1 Management of a Diffserv-based Network

QoS provisioning in DiffServ networks involves a range of management operations, from traffic engineering and admission control, to dynamic management of resources. Several frameworks have been proposed for this purpose that mainly stemmed from European collaborative research projects including TEQUILA [27], MESCAL [13], and ENTHRONE [3]. All frameworks propose the use of a general model, where the QoS management goals are realized by three distinct functional blocks: *Service Management*, *Traffic Engineering*, and *Policy Management*. The first is responsible for agreeing the customers' or peer domain's QoS requirements in terms of *Service Level Specifications* (SLSs). Traffic Engineering fulfills contracted SLSs by deriving network configuration. Policy Management provides the two aforementioned blocks with a set of policies that guide their functional behavior to reflect the high-level goals and objectives.

The business-driven approach presented in this work focuses on the functionality provided by the Service Management block. This is realized by the *SLS-Subscription (SLS-S)* and

SLS-Invocation (SLS-I) components. The former has a centralized off-line functionality and performs admission control on subscription requests based on resource availability provided by the Traffic Engineering system, whereas SLS-I is distributed across ingress routers and performs dynamic invocation of already subscribed SLSs based on the network runtime state, following operational guidelines provided by SLS-S. Both components assume an MPLS-enabled network.

3.1.1.1 Policies for Service Subscription Control

The policy-based subscription logic performs static admission control of the number and type of service subscriptions in order to avoid network overloading while maximizing subscribed traffic. The policies to achieve this objective [7][16] are shown in Table 3. The notation adopted to represent these policy actions consist of the name of the action and two parameters within parenthesis. The first parameter points the generic entity targeted by that action, namely a traffic trunk (TT) or a QoS class (QC). The second parameter refers to the parameter set by that action.

The first two actions, P1.1 and P1.2, use the notion of service satisfaction and quality levels [21] to set the relevant parameters per QoS Class (QC). Satisfaction parameters define two factors FctrAS and FctrFS $\in$ [0,0.5] to derive the rates at which a service is considered almost satisfactory (AS) or fully satisfactory (FS). They operate on an initial average rate (SRAVG) per service type that is set by the provider. An increasing AS factor produces a decrease in the service rate as SRAVG*(1-FctrAS). An increasing FS factor results in an increased service rate as SRAVG*(1+FctrAS). The Overall Quality Level parameter OQL $\in$ [-1,1] is analogous to the confidence level with which a service enjoys the agreed QoS. High priority QCs have an OQL closer to 1.

Table 3: Subscription Policy Actions

ID	Policy action	Description
P1.1	set(QC, FctrFS)	Sets the almost and full satisfaction factors per QoS class (QC)
P1.2	set(QC, OQL)	Sets the overall quality level per QoS class (QC)
P1.3	set(TT, SU)	Sets the subscription upper limit per traffic trunk (TT)
P1.4	set(QC, TD)	Sets the admission control strategy for accept/reject decision per QoS class (QC)

To describe policy action P1.3 lets introduce the Resource Availability Buffer (RAB). The RAB is a metric associated to each traffic trunk to give idea of the availability of resources in that traffic trunk. Figure 6 shows that metric consisting of three thresholds R^a_{min}, R^w_{min} and R_{max} expressed in rate units. The idea is that as close to R_{max} we set the exploitation rate of the traffic trunk much higher is the probability of not being able to provide the QoS guarantees [21]. We postpone the calculation of these thresholds to a case example in Section 3.1.3.

Policy action P1.3 (Table 3) sets the upper limit for accepting new subscriptions—*Subscription Upper* (SU)—somewhere between R^w_{min} and R_{max} and defines the level of associated risk in satisfying the QoS requirements.

Policy action P1.4 sets the Traffic Demand (TD) for a traffic trunk. Upon a new subscription request, the overall (TD) for a traffic trunk is updated by that policy action P1.4 (Table 3) taking into account the rate of the candidate service. The subscription is accepted only if TD ≤ SU in the RAB.

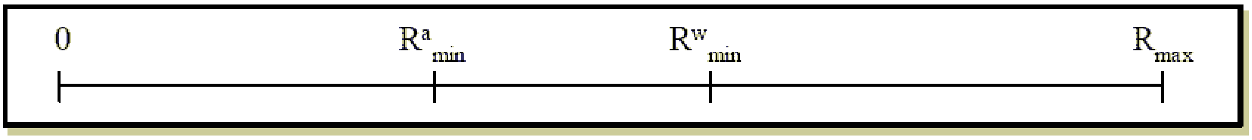


Figure 6: Resource Availability Buffer (RAB)

3.1.1.2 Policies for Admission Control

The policies used here are shown in Table 4 and are enforced on SLS-I. They perform dynamic admission control on the number of active services, as well as on the volume of admitted traffic.

Table 4: Traffic Admission Policy Actions

ID	Policy action	Description
P2.1	set(TT, TCL)	Sets the target critical level threshold wrt RAB per traffic trunk
P2.2	set(TT, SR)	Sets the service rate of a traffic trunk: SR $\in$ [AS, FS]
P2.3	set(TT, AC)	Sets the admission control threshold wrt RAB per traffic trunk: AC$\in$[R^a_{min} , R_{max}]

Policy action P2.1 defines a threshold that sets the Target Critical Level (TCL) of traffic flowing into the network, the value of which is in the range between R^a_{min} and R_{max} of the RAB. The run-time operation of a traffic trunk is triggered by TCL crossing alarms, which activate policy actions P2.2 and P2.3 for adjusting the Service Rate (SR) and the dynamic Admission Control Threshold (AC) of that traffic trunk. AC$\in$[R^a_{min} , R_{max}] controls invocations of already subscribed services. The lower it is, the less the chances are of an incoming service being successfully invoked. A new service request will be accepted only if the current utilization of the relevant traffic trunk together with the average rate of that service does not exceed AC. On the other hand, SR $\in$ [AS, FS] denotes the level of satisfaction allocated to a traffic trunk.

Note that the closer is the TCL to R^a_{min} the earlier a notification is issued, whereas values close to R_{max} result in delayed proactive actions.

3.1.2 Final System Design

The purpose of this section is to derive the above mentioned policy parameters as a function of different business indicators. For that reason we establish a model linking policies, global objectives and business indicators. On that model we reason empirically to derive the relationships as follows.

3.1.2.1 Qualitative model for Admission Control

Figure 7 proposes a model of relationships between enforceable policies global system objectives and business indicators. Reading from right to left we can say that to improve profit we have to act on the volume of subscriptions (more volume, more profit) and also on the quality (higher quality levels will lead to more profit). The volume of subscription will be determined by the parameters SU and TD of policies P1.3 and P1.4. A similar reasoning can be derived for the other entities.

In this model the importance that the service provider is wishing to give to each business indicator is expressed in terms of weights w1 and w2. If profit is priority, then w1 will be high

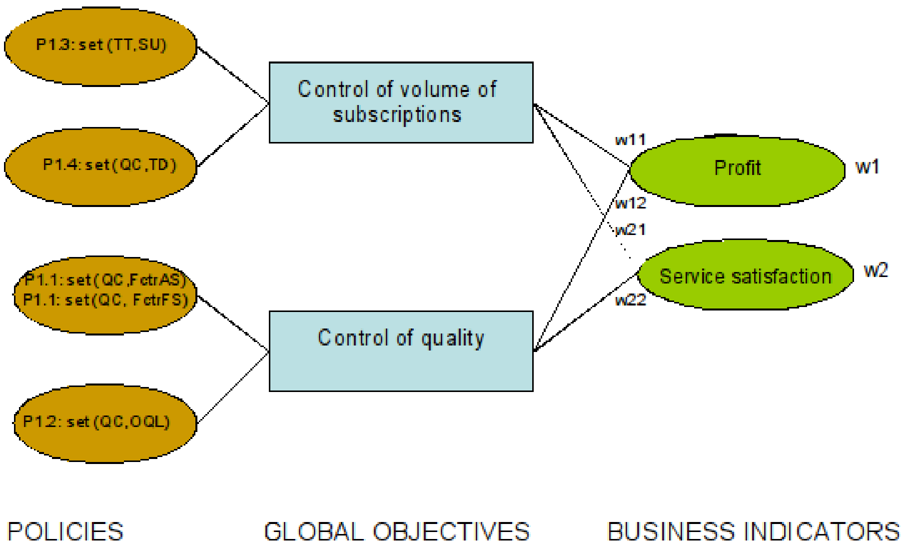


Figure 7: Model of Business Indicators, Global Objectives and Policies for Admission Control

and w2 low. In addition, each of the above weights can be decomposed in sub-weights, namely wi1 and wi2. Continuing with the example, the administrator can decide to achieve the expected Profit impact giving more or less importance to the Control of volume of subscriptions or to the Control of the quality. This will be reflected in the values that he will assign to w11 and w12.

Unless otherwise specified, these weights can be set between 0 and 1. As an example, lets envisage three scenarios:

- Scenario_1: the administrator is wishing to give much more importance to Profit than to Service satisfaction. Then w1=1, w2=0, w11=1, w12=1, w21=0, w22=0
- Scenario_2: the administrator is wishing to give the same importance to Profit and to Service satisfaction. In this case all weights are 0,5
- Scenario_3: the administrator is wishing to give more importance to Service satisfaction, Then w1=0, w2=1, w11=0, w12=0, w21=1, w22=1

3.1.2.2 Proposed Quantitative Relationships

In respect to the SU limit, setting it close to R_{max} can maximize the subscription volume. This can result in high profit but also in eventual network congestion—due to over-subscription—which can compromise service satisfaction. SU values close to R^{W}_{min} imply minimal risk of congestion but less profit due to lower subscription volumes. This is reflected in the following formula:

$$SU = R^{w}_{\min} + \frac{1 + W_{11} - W_{21}}{2}(R_{\max} - R^{w}_{\min}) \quad (1)$$

Observe that in Scenario_1 SU=R_{max}, in Scenario_2 SU=(R_{max} + R^{W}_{min})/2 and in Scenario_3 SU= R^{W}_{min}

The *Control of Volume of Subscriptions* objective is also achieved by the policy that sets the TD parameter to be used in the static admission control decision. If TD is set to its maximum (conservative approach) profit suffers, since fewer subscriptions can be accepted, whereas higher satisfaction is achieved because there is confidence that

services will be provided at their FS rates. The setting of TD to its minimum (less conservative approach) however, will result in higher profit, but lower satisfaction since there is confidence that services will only be provided at their AS rates. We define TD as follows:

$$TD = \frac{W_{11} R^{w}_{\min} + W_{21} R^{w}_{\max}}{W_{11} + W_{21}} \quad (2)$$

The two business indicators also affect the policies achieving the *Control of Quality* objective. AS factors close to 0 and FS factors close to 0.5 imply increased service satisfaction due to the higher rates produced, but can result in lower profit as fewer service subscriptions can be accommodated. The opposite results are achieved with AS factors close to 0.5 and FS factors close to 0 because of lower service rates.

$$Fctr_{AS} = \frac{1}{4}(1 + W_{12} - W_{22}) \ (3)$$

$$Fctr_{FS} = \frac{1}{4}(1 - W_{12} + W_{22}) \ (4)$$

Coherently with the above reasoning, observe that in Scenario_1 the factor AS=0.5 and FS=0 whereas the reverse occurs in Scenario_3. In case of Scenario_2 both factors are set to 0.25.

Policy P1.2 sets the OQL parameter. Values close to 1 suggest that services are provided at their FS rates, even during congestion. This results in higher satisfaction but lower profit since a less number of services can be accommodated within a given RAB. Satisfaction is reduced with OQL values close to 0 as services are most likely to be provided at their AS rates, but an increased profit is achieved with more accepted subscriptions. OQL values close to -1 result in services being provided with no guarantees. Satisfaction is at the lowest possible level whereas profit is at the highest. Therefore the following relationship is proposed:

$$OQL = (W_{22} - W_{12}) \ (5)$$

Accordingly to the above, in Scenario_1 OQL=-1, in Scenario_3 OQL=1 and in Scenario_2 OQL=0.

3.1.2.3 Qualitative Model for Traffic Control

In this model we play with three business indicators, namely losses due to rejection of service invocations (*Losses by rejection*), losses due to service performance degradation (*Losses by performance degradation*) and the *Service satisfaction* already used in the admission control model. These BIs will be put in place to achieve three global objectives, namely the Control of New Service Invocations, the Control of QoS Degradation and the Control of Rate of Active Services. These three objectives are fulfilled through the setting of the policy parameters P_{23}, P_{21} and P_{22} respectively. The way in which we propose that these BIs impact the global objectives and policy action is depicted in Figure 8.

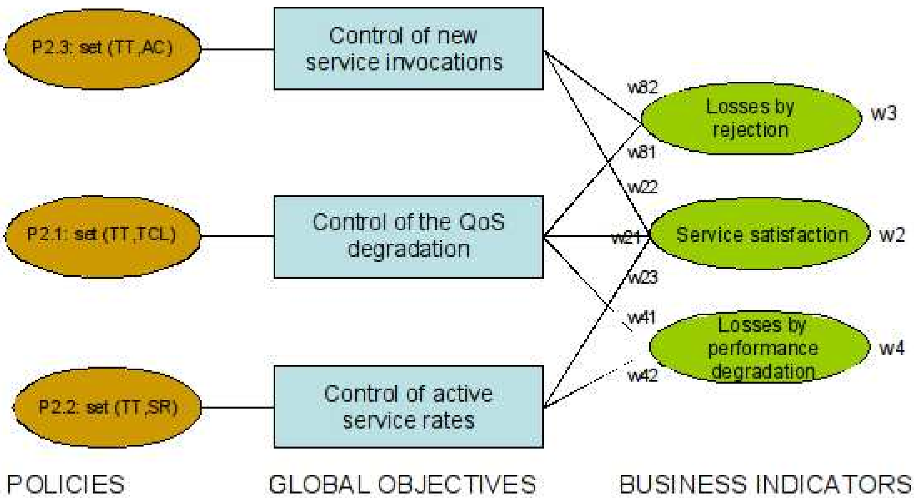


Figure 8: Model of Business Indicators, Global Objectives and Policies for Traffic Control

3.1.2.4 Proposed Quantitative Relationships

The value of TCL is an average of three values given in the formula (6) to (11) TCL_1, TCL_2 and TCL_3 that take into account the incidence of the BIs affecting this parameter.

$$TCL_1 = R^a_{min} + 2W_{21}(R^w_{min} - R^a_{min}), \text{ when } W_{21} \leq 0.5 \quad (6)$$

$$TCL_1 = (2R^w_{min} - R_{max}) + 2W_{21}(R_{max} - R^w_{min}), \text{ when } W_{21} > 0.5 \quad (7)$$

$$TCL_2 = R^a_{min} + 2W_{31}(R^w_{min} - R^a_{min}), \text{ when } W_{31} \leq 0.5 \quad (8)$$

$$TCL_2 = (2R^w_{min} - R_{max}) + 2W_{31}(R_{max} - R^w_{min}), \; W_{31} > 0.5 \quad (9)$$

$$TCL_3 = R^a_{min} + W_{41}(R^w_{min} - R^a_{min}), \text{ when } W_{21} > 0.5 \quad (10)$$

$$TCL_3 = R_{max} - W_{41}(R_{max} - R^w_{min}), \text{ when } W_{21} \leq 0.5 \quad (11)$$

Starting with W_{21}, the effect of Service satisfaction on the Control of QoS, it is reasonable that when W_{21} grows (we prioritize satisfaction in terms of QoS) the threshold TCL has to grow as well in order to prevent the decrease of the service rate due to an early crossing of the TCL threshold. In addition, we force that TCL is Ramin when W_{21} is 0.5. For these reason, the mapping function is described by formulas (6) and (7).

The weight W_{31} reflects the importance of Losses by rejection. If this BI has to be important (growing value of W_{31}) is clear that the TCL has be set high as well to avoid traffic rejection. The mapping function is the same as above.

Losses by performance degradation will also impact the setting of TCL. If we increase the value of W_{41} we could argue that TCL has to be increased in order to avoid early traffic rate reduction. But at the same tome we could argue that it has to be decreased to avoid QoS degradation due to a too high level of traffic admission. This apparent contradiction can be avoided admitting the two trends as a function of the value of W_{21} (service satisfaction). For instance, if service satisfaction has to be important then is better to decrease the TCL to avoid too much traffic. The threshold for W_{21} has been arbitrarily set to 0.5

Similar reasoning would lead to establish mapping functions between the relevant business indicators and the parameters AC and SR. AC is set between R^a_{min} and R_{max} as a linear

function of W_{22} and W_{32}, whereas SR varies between SR_{AS} and SR_{FS} as a linear function of W_{23} and W_{42}. All this is reflected in formulae (12) to (17).

$$AC_1 = R^a_{\min} + 2W_{22}(R^w_{\min} - R^a_{\min}), \text{ when } W_{22} \leq 0.5 \quad (12)$$
$$AC_1 = (2R^w_{\min} - R_{\max}) + 2W_{22}(R_{\max} - R^w_{\min}), \text{ otherwise} \quad (13)$$
$$AC_2 = R^a_{\min} + 2W_{32}(R^w_{\min} - R^a_{\min}), \text{ when } W_{32} \leq 0.5 \quad (14)$$
$$AC_2 = (2R^w_{\min} - R_{\max}) + 2W_{32}(R_{\max} - R^w_{\min}), W_{32} > 0.5 \quad (15)$$

$$SR_1 = SR_{AS} + W_{23}(SR_{FS} - SR_{AS}) \quad (16)$$
$$SR_2 = SR_{AS} + W_{42}(SR_{FS} - SR_{AS}) \quad (17)$$

3.1.3 System Evaluation

Consider a network and service provider that implements the business-driven DiffServ management approach described in this proposal, offering SLAs with bundles of FTP, web browsing and e-mail services. The provider classifies services into medium and high quality, assigning them AF1 and AF4 QoS Class respectively. For illustrative purposes we restrict our analysis to a few SLAs. Consider 7 SLAs of medium quality services with average rate 3,43Mbps each (24Mbps total avg rate) and 7 SLAs of high quality services with average rate 1,714Mbps each (12Mbps total avg rate). Consider the network topology shown in Figure 9 where ingress/egress points for medium quality services are LER1/LER4 and for high quality services LER2/LER5 respectively. Medium quality services users are located in sites 1 to 7, sending/ receiving traffic to/from site 22. High quality services users are located in sites 11 to 17, sending/receiving traffic to/from site 23.

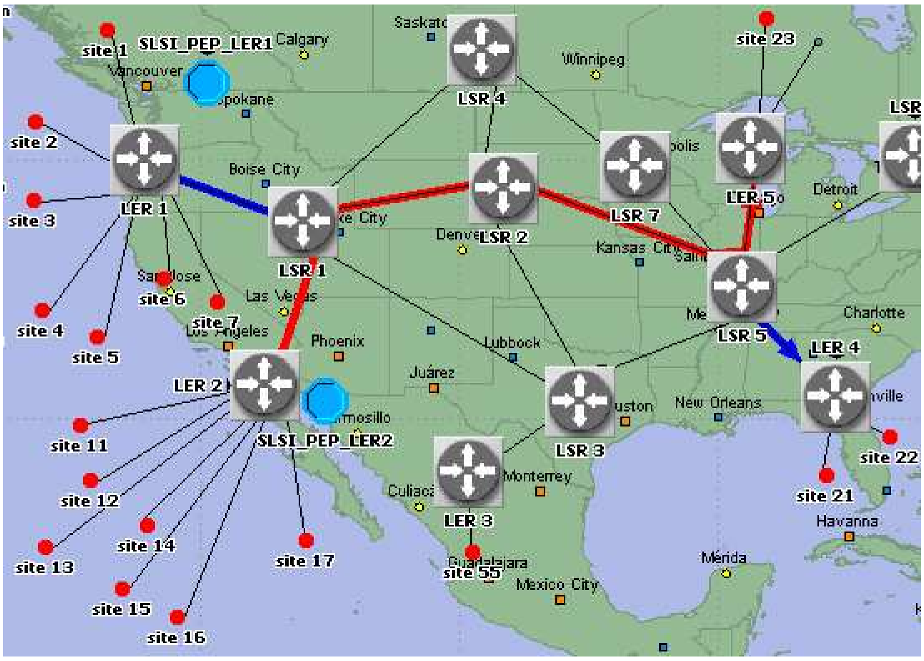


Figure 9: Network Topology for Experimental Analysis

The above information is passed to a dimensioning process resulting to the creation of two Traffic Trunks [ingress/egress, QC, RABTT], namely TT1=[LER1/LER4, AF1, RABTT1] and TT2=[LER2/LER5, AF4, RABTT2]. Following our previous definitions, the calculation of the RAB values R^a_{min}, R^w_{min}, R_{max} considers Almost Satisfied (MF_{AS}) and Fully Satisfied (MF_{FS}) multiplexing factors and the average rates of the offered services. The provider defines MF_{AS}=0.3, and MF_{FS}=0.2 for medium quality services (served through TT1), and MF_{AS}=0.4, and MF_{FS}=0.3 for high quality services (served through TT2). The Traffic Demands are calculated with the following expressions: TD_{minTT}= (1-MF_{AS})(Total avg rate) and TD_{maxTT}=(1+MF_{FS})(Total avg rate). This results to TD_{minTT1}=R^a_{minTT1}=16.8Mbps, TD_{minTT2}=R^a_{minTT2}=7.2Mbps, TD_{maxTT1}=28.8Mbps, and TD_{maxTT2} =15.6Mbps.

The value of R^w_{min} is determined after R^a_{min} has been defined for TTs sharing links. In this topology, TT1 and TT2 share links LSR1-LSR2 and LSR2-LSR5, which are standard E3 links with capacity 34.38 Mbps and represent the maximum resources available along the path for both TTs. To determine R^w_{min} we first determine the available spare resources after the sum of R^a_{min} of the TTs has been subtracted from the capacity of the shared link, in this case the spare resources are 10,37Mbps. These resources are split between TT1 and TT2 proportionally to their TD_{max}, namely 6.9Mbps and 3.5 Mbps respectively. Consequently R^w_{minTT1} = 6.9Mbps + R^a_{minTT1} = 23.7Mbps. Similarly for TT2, R^w_{minTT2} = 10.6Mbps.

Finally, R_{max} in a TT is calculated by subtracting the sum of R^a_{min} for the TTs sharing resources with the TT, from the capacity of the shared link, namely R_{maxTT1} = 34.38Mbps - 7.2Mbps = 27.1Mbps. Similarly for TT2 R_{maxTT2} = 17.5Mbps.

Now lets consider the enforcement of business-driven policies due to dynamic fluctuations of traffic. Consider that invocations of bundled services include the patterns of applications shown in Table 5.

Table 5: Patterns of Applications per Bundled Service

App	Variable	Distribution	Param.	Bundled Service
http	Page inter-arrival	Poisson	10sec	8/High Qty Svc
	Object Size	Constant	1kB	
	Image	uniform	2kB-10kB	
FTP	Inter-Request time	Poisson	15sec	8/High Qty Svc
	File Size	Poisson	500kB	
email	Send Inter-arrival time	Poisson	15 secs	8/High Qty Svc
	E-mail Size	Poisson	4kB	

Consider the provider wants to put strong emphasis on losses due to invocation rejections and losses due to performance degradation for which he assigns a value of 0,75 to BIs W3 and W4. The provider can tolerate medium levels of service satisfaction for which he assigns a weight of 0.5 to W2.

Policy parameters are derived making use of the mapping functions elaborated in Section 4.1.2, and the RAB values as described above. For example, expressions (6) to (11) are used to derive the TCL parameter values. Provided that W2=0.5, TCL1 is calculated with expression (6) (W21≤0.5). For TT1, the computation of TCL_1 gives 23,7Mbps. TCL_2 gives 25.44Mbps and TCL_3 results to 24.5Mbps. Taking the arithmetic mean of the three TCL values, we derive the TCL value of 24.5Mbps for TT1. The derivation of all policy parameters based on the mapping functions proposed above results to the policies summarized in Table 6, which are enforced in routers LER1 and LER2 in Figure 9.

Table 6: SLS-I Policy Actions

Indicators	Policy Actions	
	setTCL(TT1, 24.5Mbps)	setTCL(TT2, 12.3Mbps)
W4=0,75	setSR(TT1, 24.3Mbps)	setSR(TT2, 12.45Mbps)
	setAC(TT1, 24. 5Mbps)	setAC(TT2, 12.3Mbps)

Taking into account the patterns of applications per bundled services (Table 5), Figure 10 shows the throughput of TT1 and TT2 respectively in the network with standard DiffServ mechanisms in 5 minutes in which the provider serves excessive invocations (more invocations than the dimensioning process estimations). The bottom part of the figure shows the amount of invocations received and the active services during each period of the scenario execution; medium quality services are invoked every 20 sec and they are enjoyed 70 sec. High quality services are invoked every 40sec and enjoyed 80 sec.

Serving excessive invocations at a time (e.g. up to 28 medium quality services, plus up to 14 high quality services) yields to network over-utilization and eventually to congestion. Without control of any kind, the service satisfaction is severely reduced for both, medium quality and high quality services. In addition, the potential penalties due to performance degradation should be very high due to the over-utilisation and congestion states of the network during most of this scenario, affecting severely the business value of the provider.

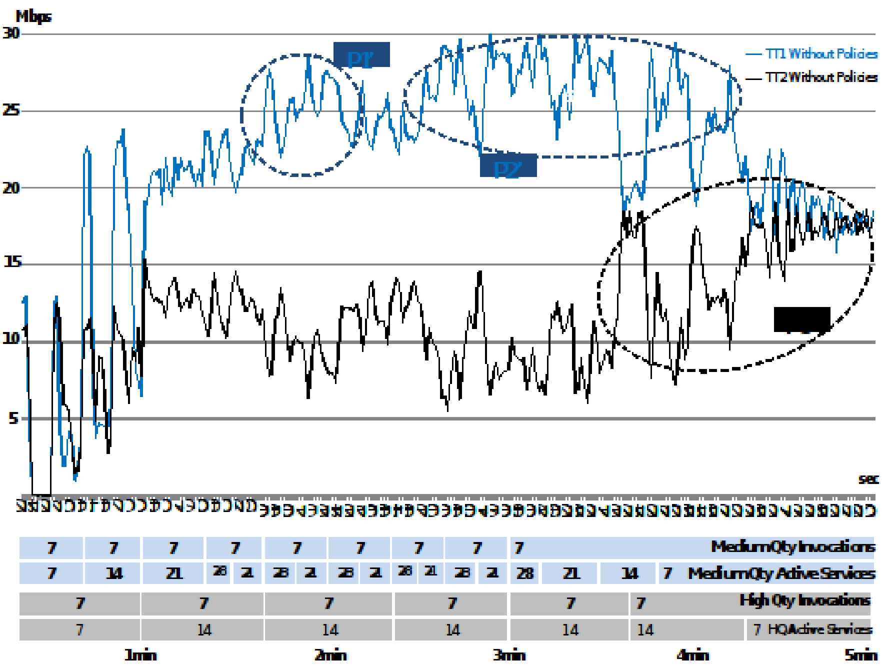


Figure 10: Scenario Execution with Standard DiffServ Mechanisms

Figure 11 shows the behavior of the network driven by the business-oriented policies introduced earlier. Even when the traffic generated in both execution runs is not exactly the same, it also follows the patterns shown in Table 5. For comparison purposes the invocation requests are the same as for the DiffServ standard execution. The actual service

invocations, active services and rejected invocations during the scenario execution are shown at the bottom of Figure 11.

The business-driven policies force the network to exploit the resources, but at the same time to protect it for eventual service degradations. For example, up to 28 medium quality services are served before sensible proactive actions are taken as a result of TCL crossings in TT1 (see P1 in Figure 11). A total of 14 invocation rejections and service rate adjustments result to a sensible reduction of the traffic injection by users, around TCL for TT1 (see P2 in Figure 11 compared to P2' in Figure 10). The system has been forced to serve between 14 to 7 medium quality services which is still highly profitable provided that the dimensioning estimations were originally set to 7 medium quality services for TT1. This way the potential losses due to invocation rejections are reduced. User satisfaction is in general acceptable as users enjoy rates close to the TCL, above R^{w}_{minTT1} (23.7Mbps). Potential losses due to performance degradation are also low. With the standard DiffServ mechanisms, the resources are over-exploited putting in danger the service satisfaction and consequently affecting the business value of the provider (see P1' and P2' in Figure 10).

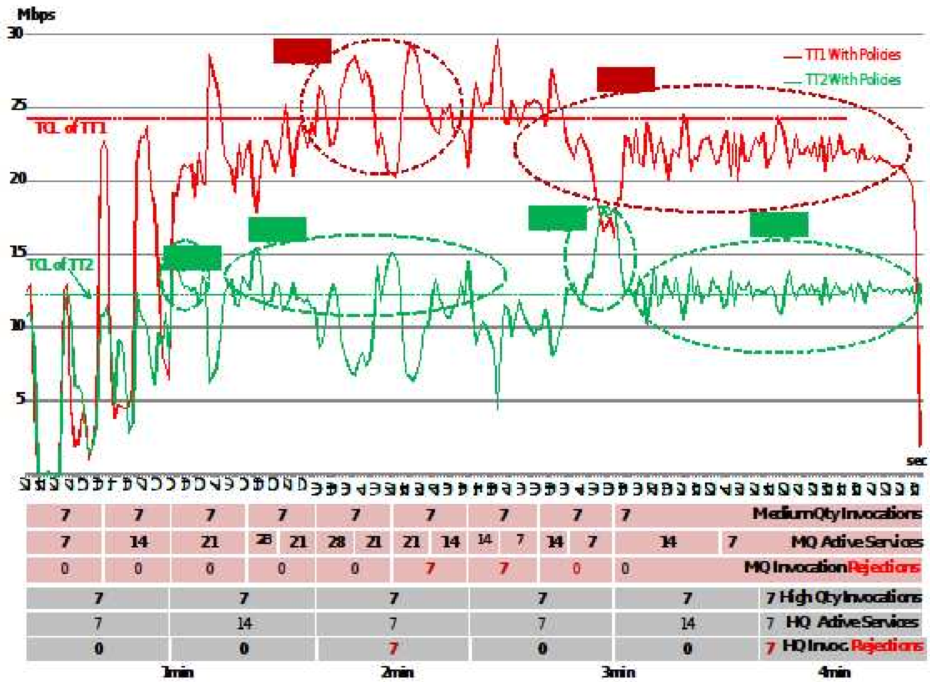


Figure 11: Scenario Execution with the Business-oriented Policies

Similar results are obtained for high quality services served through TT2. An interesting situation in TT2 is presented as a result of the enforcement of TT1's policies. Due to the reduction of traffic injection in TT1, the DiffServ mechanisms allow traffic served through TT2 to be slightly increased, see for example P4 and P5 in Figure 11. The same principle is applied in these conditions, proactive actions are taken when traffic goes beyond TCL for TT2, resulting to 14 invocation rejections of high quality services along the scenario execution and the appropriate service rate adjustments, resulting in an eventual normalization of traffic injection in TT2 (see P3 in Figure 11). This way the sensible increase of traffic injection exhibited in the normal DiffServ scenario are avoided (P3' in Figure 10). The standard DiffServ mechanisms are concealed with the business oriented approach applied here to increase the business value of the network, serving between 7 to 14 high quality services and taking care of the quality of the services.

For demonstration purposes, the scenarios described above have been executed disabling the traffic shaping mechanisms of the edge routers. This way the traffic fluctuations have been made more evident to show the effectiveness of the approach. In addition, in order to avoid instability due to very short statistical traffic fluctuations, we have used a time window of 5 seconds for the TCL-crossings. This is exemplified in P4 in Figure 11 where the actual policies are not triggered therein as the TCL crossing is not longer than 5 seconds. In systems where traffic shaping mechanisms are mandatory, higher time windows may be necessary to be defined.

3.2 Pricing Mechanism for Wireless Access Networks Management

Our cities are already covered by a myriad of diverse wireless access networks. The most ubiquitous are the well organized homogeneous and centralized operator-based cellular networks that base their business on a captive client. However, a more heterogeneous and free network of WiFi, WiMAX and 3G access-providers has appeared and it is starting to challenge the dominant billing paradigm. In this new world, clients can choose the best or the cheapest provider for the time and place they are trying to connect from, and a free market of Internet connectivity providers is emerging.

In this context, pricing will have a fundamental role. Using the right pricing strategy, an operator will try to obtain the highest possible revenue while the users will try to get a service that fits their requirements at the minimum possible price. As stated in [22]: *“From an economic point of view, pricing plays an important role in trading any resource or service. The most important objective of trading is to provide benefits to both the sellers and the buyers. Therefore, the price must be chosen so that the revenue of the sellers is maximized while the highest satisfaction is achieved by the buyers.“*

This work presents a distributed, rule-based pricing system that implements exactly the same intuitive ideas in the shape of policy-rules to be enforced on the price charged by each provider. Those rules are aimed to improve the quality of service and to increase the global income of a service provider in that world in which users are free to choose every time they connect.

3.2.1 Pricing Mechanism to Fulfil Network Load Distribution and QoS Goals

The rationale behind the proposal in this paper is twofold: first, it tries to be an autonomic, scalable pricing system that provides operators with simple and business-understandable means to set prices following the principles of demand and competence mentioned above. Second, our proposal is aimed to provide users with a means to be confident that the service they choose is the best in terms of quality and price among all that are available in a given coverage area.

3.2.1.1 Business Model

We envisage scenarios where three types of actors coexist on a geographical area: *users*, *providers* and a *regulator* (see Figure 12). *Users* are persons in possession of some wireless-capable device that are willing to establish a connection to the Internet with some quality of service requirements at the minimum cost. In our model, if the services provided are substitutable, the users will choose always the less expensive provider. The *providers* have an access network conformed for a set of access devices that we will call generically Access Points (AP). A *provider’s* objective is to sell access services to users while maximizing their revenue. The third entity, the *regulator*, is a neutral entity, probably played

by a governmental agency in a real setup, whose objective is to enforce the sharing of pricing information between the providers.

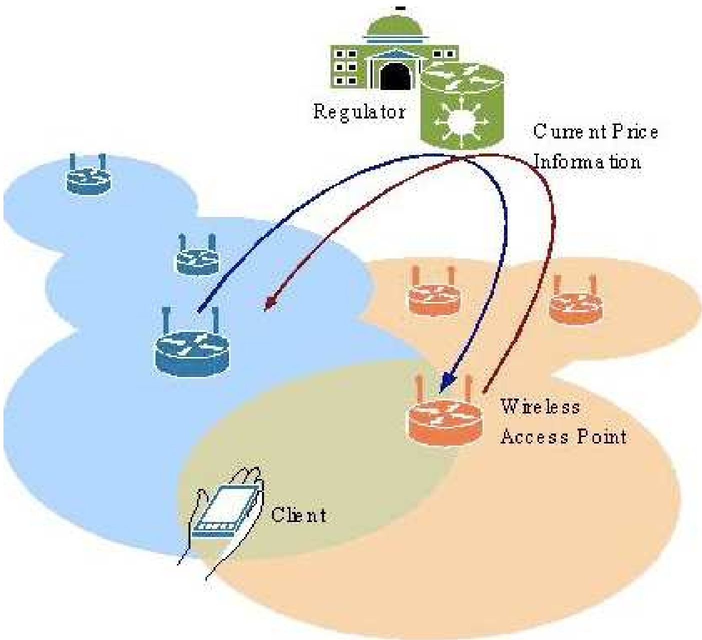


Figure 12: Involved Actors in the Proposed Application Scenarios

3.2.1.2 Governing Policies

Policies are intended to allow each AP to decide which is the most advantageous price for its own connectivity service, having into account its context, user's demand and potential competitors. These decisions are made by a Policy Decision Point [20] installed in the same AP, independently from the others regardless whether they belong to the same or competing providers, following a set of policy-rules modeling the economic criteria of demand and competence mentioned above. It is worthy to mention that these sets of policies can be freely established by each provider according to best practices founded on past experience, forecasts based on economic models, etc and that by no means are imposed by the regulator. Even more, a provider could participate without making use of

Table 7: Rules Driving an Access Point Behaviour in Front of Service Demand

Rule	Description
Rule 1	**if** few_users **and** users_steady **then** decrease_price_slow
Rule 2	**if** few_users **and** users_decreasing_slow **then** decrease_price_slow
Rule 3	**if** few_users **and** users_decreasing_fast **then** decrease_price_fast
Rule 4	**if** few_users **and** users_increasing_slow **then** keep_price
Rule 5	**if** few_users **and** users_increasing_fast **then** keep_price
Rule 6	**if** mid_users **and** users_steady **then** increase_price_slow
Rule 7	**if** mid_users **and** users_decreasing_slow **then** decrease_price_slow
Rule 8	**if** mid_users **and** users_decreasing_fast **then** decrease_price_fast
Rule 9	**if** mid_users **and** users_increasing_slow **then** keep_price
Rule 10	**if** mid_users **and** users_increasing_fast **then** increase_price_slow
Rule 11	**if** lots_users **and** users_steady **then** keep_price
Rule 12	**if** lots_users **and** users_decreasing_slow **then** decrease_price_slow
Rule 13	**if** lots_users **and** users_decreasing_fast **then** decrease_price_fast
Rule 14	**if** lots_users **and** users_increasing_slow **then** increase_price_slow
Rule 15	**if** lots_users **and** users_increasing_fast **then** increase_price_fast

pricing policies, just fixing a flat rate for its services. Nevertheless, for the sake of illustrating the approach and also to allow a quantitative evaluation we present hereafter a specific sets of policies; one is concerning the demand and the other is towards the competition.

The set of rules driving the decisions regarding demand is presented in Table 7. The rationale behind this set of rules is simple; the price of the service is increased, kept constant or decreased depending on the number of users served and its gradient of change. In this way, the price will be adapted to stimulate or inhibit service demand and its adaptation rate will track the evolution of such demand. In practice we accomplish it classifying the number of users in three categories (few, mid and lots), the gradient of change in two (slow and fast) and also allowing two rates of price change (slow and fast).

On the other hand, the set of rules facing the competence is presented in Table 8. Here the global objective is to accommodate the price to the evolution of the competitors to avoid users' migration. In particular we have classified the price of the competence in two categories (lower and higher) and considered two adaptation rates (slow and fast). In this context, a competitor is an individual AP and two APs are competitors if their coverage regions overlap and they belong to different providers.

Table 8: Rules Driving an Access Point Behaviour in Front of Competitors' Price

Rule	Description
Rule 1	**if** competitor_price_lower **and** _price_decreasing_slow **then** _price_slow
Rule 2	**if** competitor_price_lower **and** competitor_price_decreasing_fast **then** decrease_price_fast
Rule 3	**if** competitor_price_lower and competitor_price_steady then decrease_price_fast
Rule 4	**if** competitor_price_lower **and** competitor_price_increasing_slow **then** decrease_price_slow
Rule 5	**if** competitor_price_lower **and** competitor_price_increasing_fast **then** decrease_price_fast
Rule 6	**if** competitor_price_higher **and** competitor_price_decreasing_fast **then** decrease_price_slow
Rule 7	**if** competitor_price_higher **and** competitor_price_steady **then** increase_price_slow
Rule 8	**if** competitor_price_higher **and** competitor_price_increasing_slow **then** increase_price_fast
Rule 9	**if** competitor_price_higher **and** competitor_price_increasing_fast **then** increase_price_fast

3.2.2 System Design

Our solution combines centralized, hierarchical and fully-distributed management structures to address different challenges with the most appropriate approach. Our basic design criterion is to maximize the distribution of tasks over the nodes as much as possible. Only the tasks that inherently require a centralized organization, such as global optimizations, or those tasks which perform better on a weakly distributed structure, are carried out using hierarchical structures. Management tasks such as local optimizations follow a fully distributed approach.

The system can be seen as mainly composed by independent units or nodes that work as peers deployed at each managed network device (a wireless access point). From a high level architectural point of view those nodes follow a classical policy-based architecture constituted by a Policy Decision Point (PDP), a Policy Enforcement Point (PEP) and an agent that monitors the state of the device and the behavior of the network. Policies follow the ECA model, where the condition may be the occurrence of some event (*e.g.*, an alarm or a service request) or certain network state, and the action is the desired response when the condition is true or just it occurs. Notifications sent by the monitoring agent of any of the nodes of the network are the source of events for the PDP of the same or any other node. All the communications between nodes, including control messages, notifications and request for actions are carried out by a content-routed asynchronous communication bus that organizes the nodes in a hierarchical overlay as seen in Figure 13. The

communications bus' design is inspired by existent notification services such as Siena [20]. Finally, a centralized management station is in charge of the edition, optimization and distribution of rules into the nodes.

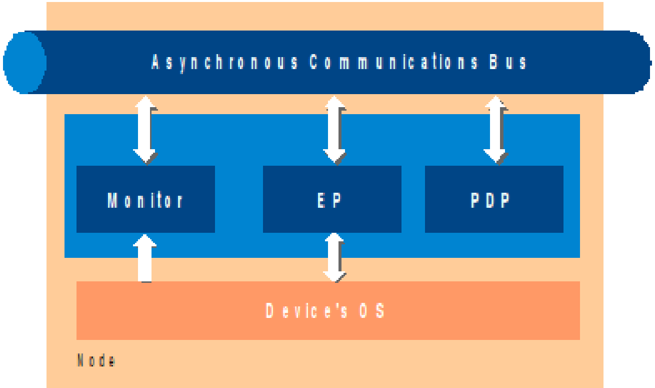


Figure 13: Involved Actors in the Proposed Application Scenarios

The reconfiguration actions decided by the PDP may also involve other network elements, in which case the node may be the manager of the activity and delegate responsibility for some actions to other nodes. As it will be described in the next section, policies have to be managed as finite state transducers. The translation from policy rules expressed in natural language into the finite state transducers implementing those rules is currently made by a programmer assisted by the algorithms presented in [5]. Despite the fact of being a software assisted process, it is still a complex and work-intensive task.

3.2.2.1 The Logics behind the Pricing Management System

Finite State Automata are classical computational devices used in a variety of large-scale applications. FSTs, in particular, are automata whose transitions are labeled with both an input and an output label. They have been useful in a wide range of fields, but particularly in Natural Language Processing. This discipline makes intensive use of grammatical rules, which are ambiguous by nature, and requires quick decisions based on those rules, in particular in fields such as speech recognition, with major performance requirements. Additionally, Finite State Machine-based solutions are typically light-weight. They can be implemented as arrays of states, transitions, and pointers among them without falling into heavy management structures.

For representing policies with FSTs we used the model presented in [2], where a more detailed description of it can be found. A policy has a condition delimiting a “region of events" where a given event can or cannot lie. When such an event is inside two or more overlapping regions a modality conflict may arise. We are concerned about how tautly a condition fits to an event instead of how far from the border it is. Thus, the preferred condition will be that which is the most “taut” around the event under consideration. In order to formally represent the aforementioned *tautness*, we use the metric called Tautness Function, a real number in the interval [-1,1] so that the tauter a condition is, the closer its TF is to zero. Examples of tautness functions are given here after.

3.2.2.2 Processes to be Executed to Use to Represent Policies as FSTs

This section describes the processes related to the policy model. Several tasks have to be performed to generate, deploy, and evaluate the FST corresponding to a policy set. These

tasks are: policy translation, conflict resolution, model deployment, context gathering, policy evaluation and tautness function computation.

Policy Translation

Policy translation from high-level languages into internal policy evaluation models can be a complex task that needs to be kept simple and performed on an ad-hoc basis in our system. Following the principles presented in [2], we translate high-level policies into the internal policy evaluation model comprised of FSTs.

Conflict Resolution

An advantage of using transducers to model policies is the rich set of operations available. We can join, intersect, complement, compose and determinize transducers under certain conditions. For example, to build a FST that models a set of positive obligation policies, we must build one for each policy and join them using the *union* operation. However, the union of FSTs maintains ambiguities and contradictions. Therefore, *determinization* and *composition* operations must be performed to eliminate these problems. Both operations are extensions to algorithms for predicate-augmented Finite State Transducers and were developed by Baliosian et al. [2].

Determinization transforms a FST into its deterministic and unambiguous version, in fact it also eliminates, when possible, *static conflicts* between policies.

If a FST is deterministic, the process of computing the output actions for a given stream of events can be implemented efficiently. This process is linear in and independent of the size of the FST. The determinization algorithm has two main stages:

1. *Eliminating apparent local conflicts.* Local ambiguity may not be such if by analysing the whole transducer, we realize that only one path is possible until the final state.
2. *Resolving static conflicts.* If it is not possible to shift output labels further, the second stage begins. A transition is created for each possible combination between potentially conflicting conditions applying the following criterion: although an event satisfies two conditions, one of these conditions fits more *tautly* than the other. The idea of tautness is represented by the Tautness Functions defined above, which can be used to compare orthogonal conditions.

In the output part of the transition, actions and events are arranged following the order given by operators on the input. These operators are in fact part of the output. Later in the process these operators will be eliminated by *composition* of transducers to apply the given constraints in the system.

Composition eliminates semantic contradictions (*i.e.*, dynamic conflicts) between the actions. This operation between transducers is equivalent to the composition of any other binary relation.

Thus, the process can be understood as a chain of events where the events and actions in the output of the first transducer are considered to be the input of the second one. The advantage is that the chain process is performed analytically in the network and not on the mobile device.

Thus, if we create a FST that replicates all input actions on the output except for those patterns of actions not allowed on the system, and then we compute the *composition* of that transducer with the FST policy model, we obtain a transducer that enforces actions without

dynamic conflicts. This means actions that must not be performed at the same time, for example two handovers, each one to a different network.

Consequently, conflict resolution is intrinsic to the model. This process not only builds the transducer that models the policies, but also eliminates ambiguities and contradictions between those rules. The main steps are:

1. Compute the union of all transducers representing *rights* and *obligations*.
2. Subtract the transducers representing *prohibitions* and *dispensations*.
3. Compose the resulting transducer for each constraint transducer.
4. Determinize the resulting transducer to solve conflicts.

Model Deployment

After policy translation, conflict resolution and FSTs composition, the final set of FSTs is built (everything so far happens on the network side). Thereafter, the FSTs need to be sent to the repository in the AP. A Model Deployment Module together with a Policy Master, are responsible for the installation process.

Policy Evaluation

Policy evaluation occurs in the FST model. As we mentioned earlier, the computational load of deterministic transducers does not depend on the size of the transducer but rather only on the length of the input. This is possible because the computation consists of following the only possible path corresponding to the input represented by an *epoch* or window of events that are considered simultaneous for the purpose of detecting dynamic conflicts.

When evaluating the epoch, the transducer performs two tasks: it checks the current epoch and decides if it contains a relevant event pattern in order to decide whether or not to accept it; then it produces a sequence of actions for every accepted epoch, which is sent to the Policy Enforcement component.

Tautness Function Computation

A fundamental process in the deployment of FST models is the appropriate computation of tautness functions. Our prototype handles each component of the demand and competence individually with the common idea of expressing the intuitive idea behind each of those components. For example, when computing the conditions *fewUsers, midUsers* and *highUsers* they are expressed somehow complementary but not completely exclusive between them.

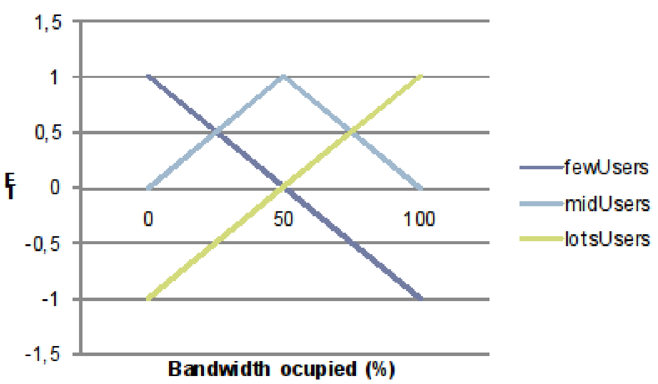


Figure 14: Tautness Functions fewUsers, midUsers and lotsUsers

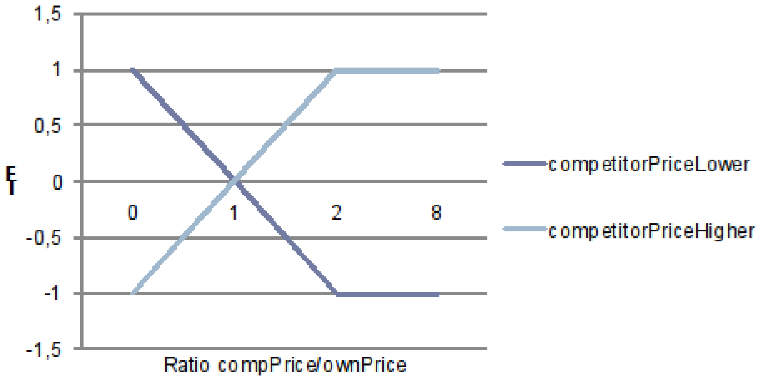


Figure 15: Tautness Functions competitorPriceLower and competitorPriceHigher

A value as close to zero as this one means a *very strong condition*. Hence, it is *very unlikely* that this situation will occur and the manager must have had a very good reason to specify a policy with this condition. Therefore, during the determinization process this condition will have a high priority. Nevertheless, at runtime each TF value will be pondered according to the user profile.

3.2.2.3 A Conflict Resolution Example

To better understand how the FST-based conflict resolution presented in Section 3.2.2.2 works, we will describe it with the help of rules slightly simpler than those presented above. Let us consider the two rules below:

Rule 1: if few_users then decrease_price_slow

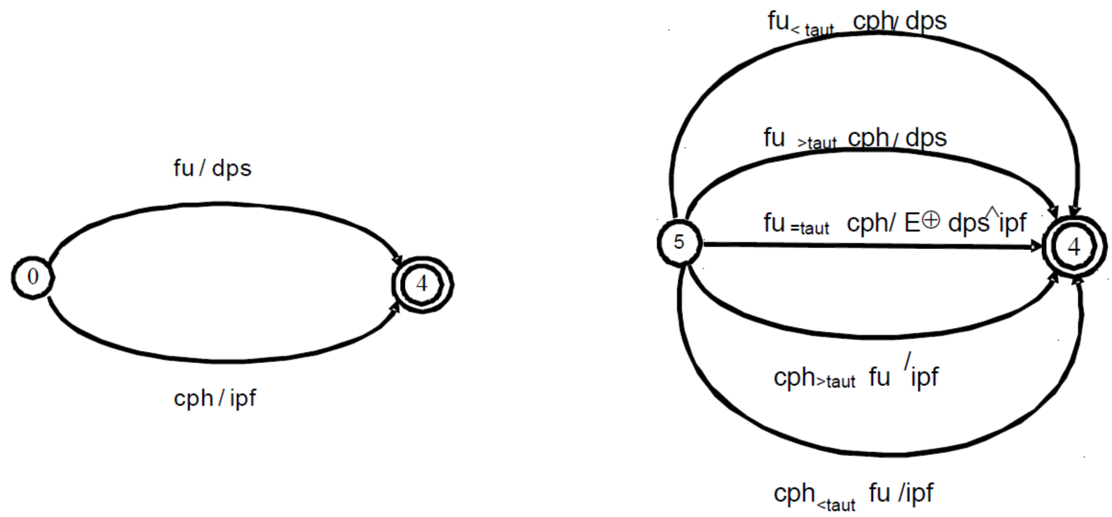
Rule 2: if competitor_price_higher then increase_price_fast

Rule 1 models a pricing policy based on demand, and rule 2 models a pricing policy based on competence. Although in a simpler manner, they are the most patent case in which our previous set of rules may have a conflict.

The FST in Figure 16(a) models both rules at once. For saving space in the figure, TF few_users (see Figure 14) is depicted as “fu”, TF competitor_price_higher (see Figure 15) as “cph” , action decrease_price_slow is depicted as “dps” and increase_price_fast as “ipf”.

For simplicity, we are considering the case in which a single notification arrives, then the PDP makes a decision based on it, and nothing else happens. (This makes the FST simpler because only has to accept strings made of a single symbol).

Now, consider the case in which a notification arrives stating that the price of a competitor has changed. At that time the PDP runs the FST in Figure 16(a) as follows: starting in the initial state “0” evaluates the TFs in the input label of each of the edges going out of the state. This is the PDP will evaluate few_users and competitor_price_higher. In the case that there are few users connected to the local AP and that the price of the competitor is lower or equal to the price charged by the local node, few_users will evaluate to a positive number and competitor_price_higher to a negative number. Therefore, the PDP chooses the upper edge of the FST, which produces decrease_price_slow as output, causing the PDP to order the local EP to decrease the charged price slowly. However, in the case that the price of the competitor is higher we have two edges going out of state “0” with positive TFs and no easy choice can be made. To avoid this problem, before being deployed in the AP’s PDP, the FST is determinized and the FST in Figure 16(b) is computed using the



(a) Before Determinization (b) After Determinization

Figure 16: Determinization of Two Simple Rules

algorithm mentioned before. Now, when the PDP evaluates the determinized transducer, at each state it has only one possible edge to follow. Let us assume that, when the PDP evaluates it, the TF few_users takes a value of 0.3 and competitor_price_higher a value of 0.5. We are in the case in which few_users is "tauter than" competitor_price_higher, expressed in Figure 16(b) as "fu >taut cph", therefore, to run the transducer produces a single output, in this case the order to decrease the price charged by the local AP slowly (action decrease_price_slow).

3.2.3 System Evaluation

The simulation platform presented hereafter was designed with the aim to evaluate the viability of the proposed solution and, above all, the viability of the new pricing paradigm. In this section we present the result of two sets of experiments in which we compare the behavior of our proposal against the most popular pricing strategy for WiFi and 3G access networks.

3.2.3.1 Evaluation Setup

In order to evaluate our solution we simulated an environment with two competing providers offering access services on the same geographical region and with identical networking infrastructure resources. However, one of the providers follows our pricing strategy and the other follows the nowadays prevailing fixed-rate pricing structure.

Although it will not happen in a real deployment, to ensure a fair comparison between competitors in our experiments, both providers have identical sets of APs installed on the same coordinates and with identical coverage areas. A large number of clients move along the area covered by both providers choosing to connect to the AP with the lower price for a randomly specified service.

For each experiment, each provider has nine APs deployed in a 1200x1200m region. Those APs are distributed in a mesh at approximately 400m of each other (see Figure 19). As said before, APs are by pairs, one owned by the provider following our strategy and the other owned by the competitor provider following a fixed-rate strategy. The price charged by each AP is a random variable that follows a uniform distribution with values between 50 and

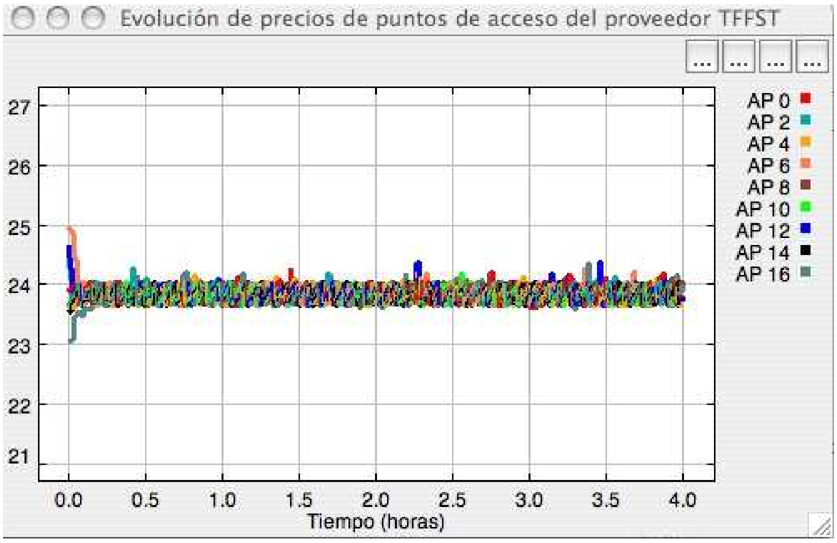


Figure 17: Prices by AP for the Provider Using our Strategy (Manhattan Mobility)

60 units. Each AP has an available bandwidth of 3.5 Mbps, and a homogeneous circular coverage area with a radius of 285m.

For the experiments, there are 450 mobile users. Each user tries to hire a connectivity service with a random bandwidth between 50 and 150 Kbps with any of the APs in its connectivity range. These users move inside the square area mentioned above following one of two mobility models.

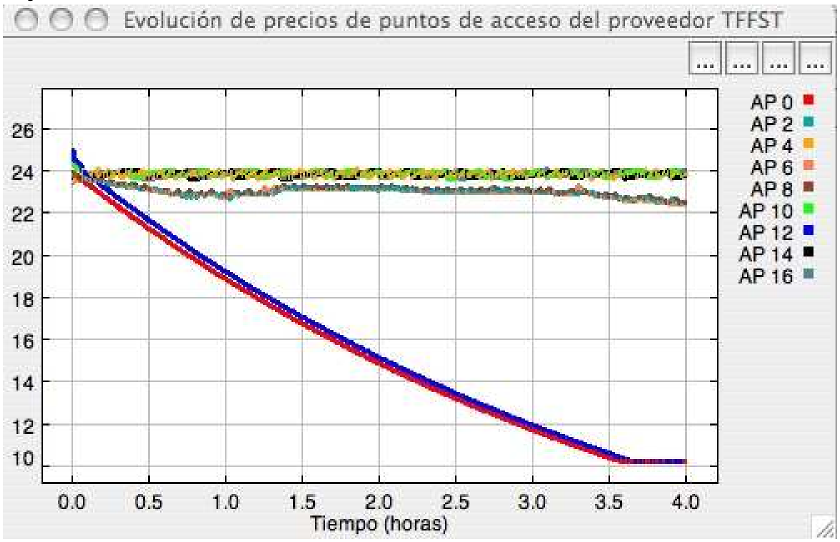


Figure 18: Prices by AP for the Provider using our Strategy (RPGM Mobility)

In the first set of experiments we assume an urban environment and use the Manhattan mobility model [9] to capture the movements of the users. Following this model, each user moves along a mesh of parallel, perpendicular streets placed every 100m. For all the experiments using the Manhattan model, the speed of a terminal varies randomly with a pedestrian mean of 1m/s and a standard deviation of 0.2. At each intersection, a terminal decides whether to turn left, right or keep the direction with a probability of 0.2.

The second set of experiments were made to study the balancing capabilities of our strategy assuming that the clients move following the *Reference Point Group Mobility* (RPGM) model [12]. The RPGM models a set of clients moving in groups but with some individual freedom to move inside and between groups. Each group has a logic center

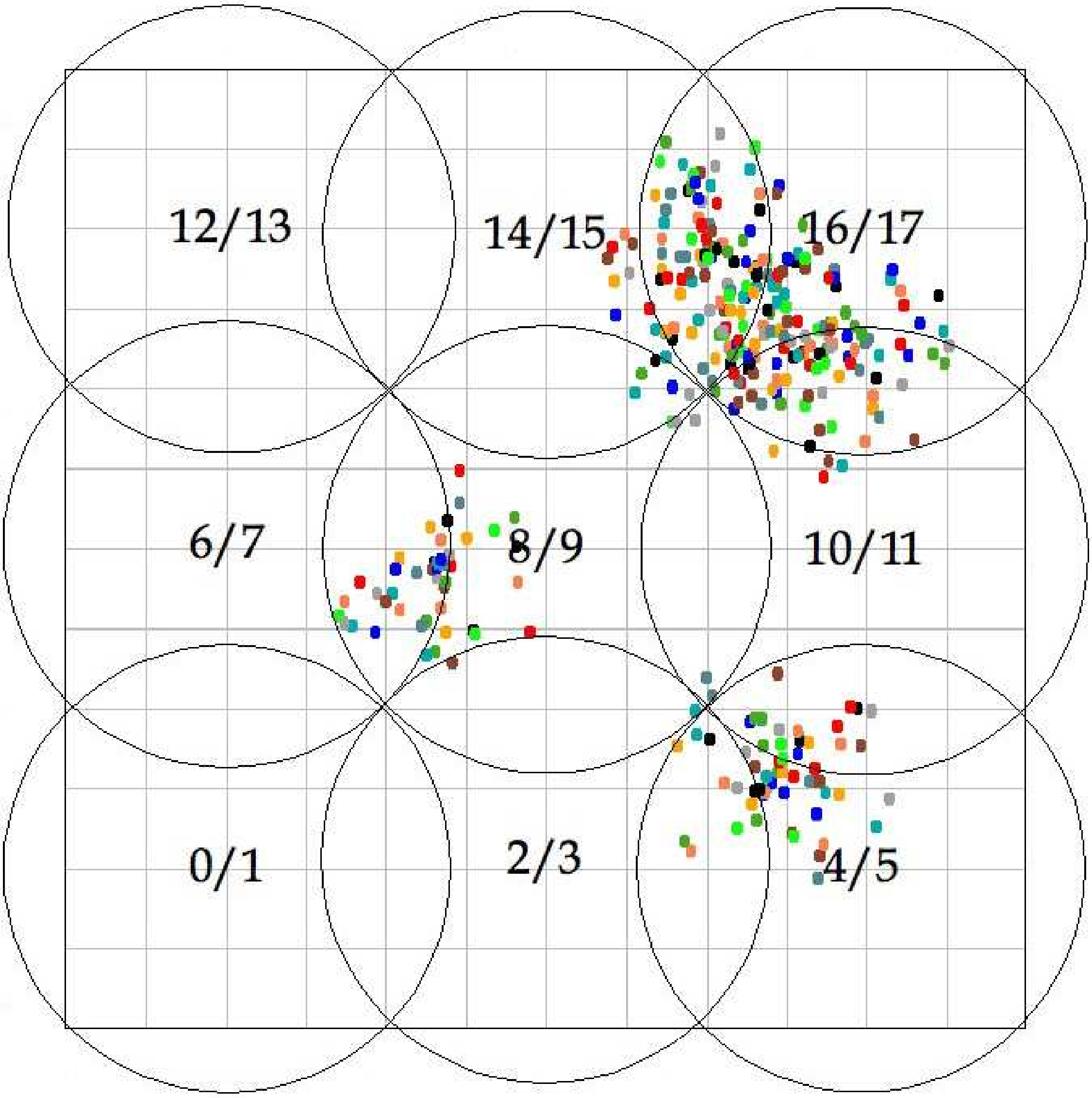


Figure 19: Position of APs and Users (RPGM Mobility)

(group leader) that determines the behavior of the rest of the group. At the beginning of the simulation each client is uniformly distributed in the vicinity of the leader, then, each client moves with random speed and direction. All the experiments run during a simulated time of 240 minutes.

The metric used to compare the behavior of both providers is the total amount of money made by each provider (adding the money made by all its APs) within the simulation time.

3.2.3.2 Evaluation Results

Figure 17 shows the price evolution for the 9 APs of the provider implementing our pricing strategy. In this experiment the users move following the Manhattan model. In the figure, it is possible to see how the prices change getting similar to the competitor price but slightly below. This makes the clients to prefer our provider (see Figure 21). The higher number of clients connected to APs owned by our provider explains the bigger revenue despite charging lower prices for the same service.

The *Reference Point Group Mobility* left regions of the geography without users and regions with high concentrations of them. This situation makes more visible the price dependence

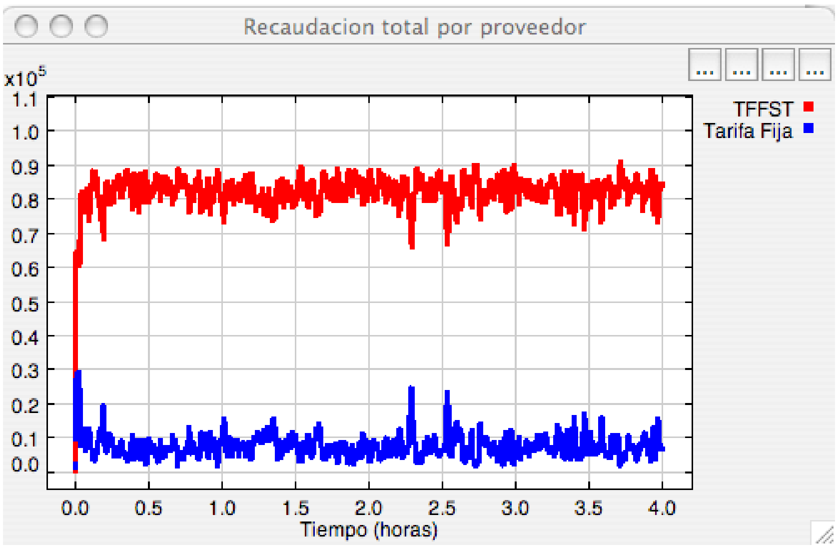


Figure 20: Revenue by Provider (Manhattan Mobility)

on the number of users trying to connect to an AP, as can be seen in Figure 18. In this experiment, APs 0 and 12 have no users in their coverage regions (see Figure 19), consequently, the price charged by those APs decreases until it reaches the minimum possible cost defined by the provider. The APs 2, 6, and 8 have a small number of users in range, therefore, they moderately reduce the price. Finally, those APs with most clients in

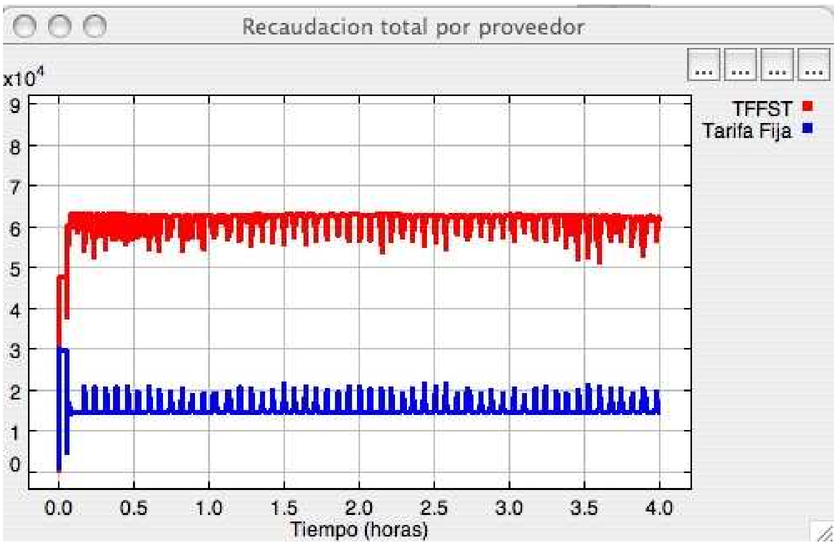


Figure 21: Revenue by Provider (RPG Mobility)

their coverage area maintain their price close to the price charged by the competitor provider. This divergence between the prices charged by different APs of the same provider causes that some clients migrate from their closest APs to an AP with a lower price. In this manner, the pricing strategy works as a network balancing tool. We can see that the global earnings made by our provider are also higher than the earnings of the competitor (see Figure 20 and Figure 21).

This experiment also shows how rule-based pricing can be used as a congestion control tool. As more users start connecting to a given AP, the price rises following Rule 15 for example naturally pushing some users out to other APs. Finally, an accessory, hard access-control prevents the connection of too many users to an AP in case the pricing mechanism is not enough.

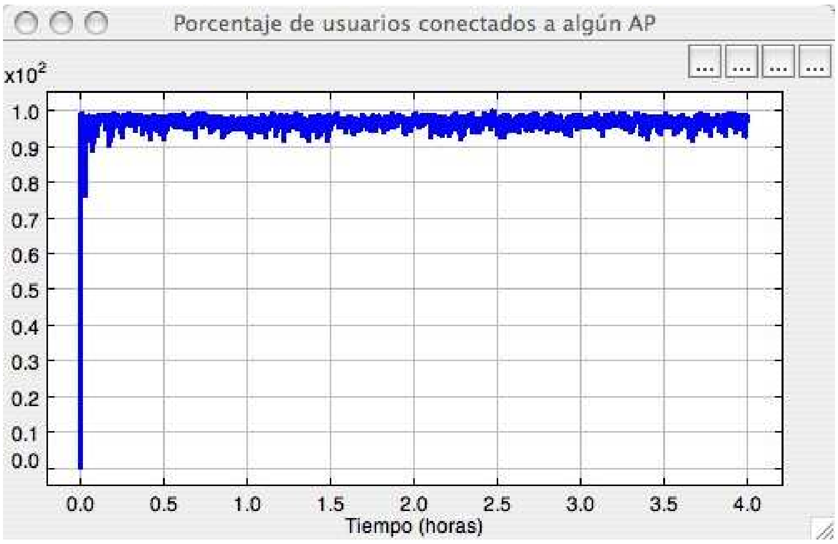


Figure 22: Percentage of Users Connected to any AP (Manhattan Mobility)

The stability of the connections is related with the users' movement and velocity. When a user that is connected to a given AP is informed of the existence of a lower price AP in its current position, it terminates its current connection and tries to connect to the AP with the lower price. However, as can be seen in Figure 22 and Figure 23, a high percentage of users are connected along the simulation to any of the AP with the only exceptions of the intervals in which the users are switching from one AP to another.

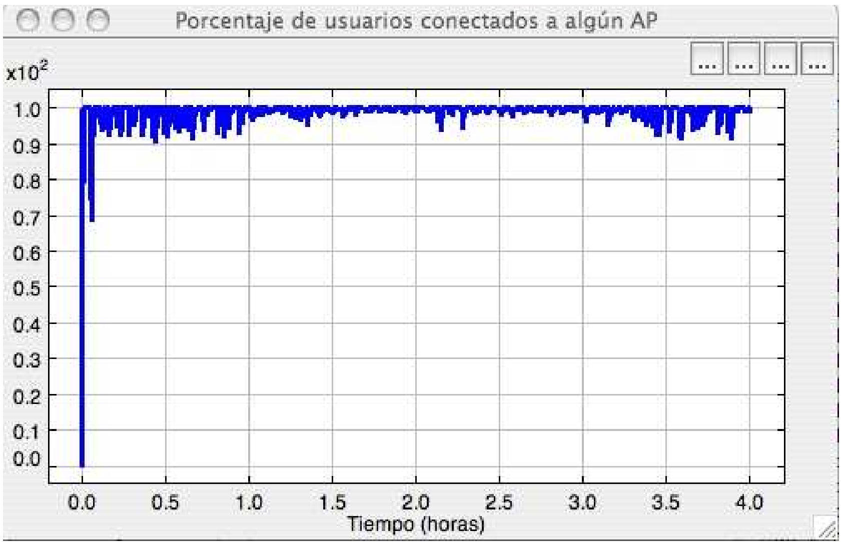


Figure 23: Percentage of Users Connected to any AP (RPG Mobility)

4 IELOS — Impact Analysis of Economic and Legal Objectives on Basic and Value-added Services

The goal of the IELOS project is to research the impact of economic and legal objectives on services provided by ISPs and value-added service providers. IELOS addresses the respective threefold point of views of the network, the provider/user, and the service. According to high level economic objectives, this involves for instance policy refinement techniques that derive the necessary service management and traffic engineering QoS provisioning policies. According to legal objectives, this addresses the automated determination of jurisdiction and applicable law as suited for a to be concluded international service contract.

The work in IELOS investigates, on the one hand, the trade-offs between profit, customer satisfaction, and service quality by considering their relationship to parameters such as the number and type of subscriptions formed, and the amount of traffic injected into the network. On the other hand, trade-offs between the effort needed and gains achieved to automatically determine jurisdiction and applicable law by the use of specific rules are assessed in this project. Moreover, potential compatibility issues arising from different national legal viewpoints are identified and integrated as far as possible in order to facilitate a single consistent, harmonised definition of key parameters of an international service contract like habitual residence and domicile of service users and providers.

IELOS addresses the set objectives by examining the impact of economic and legal factors on the management of commercial services as follows:

- Specification of business indicators and their relationships with low-level network configuration policies.
- Optimization of network and service management functionality to meet economic objectives.
- Functional evaluation of the approach based on a simulated DiffServ environment using OPNET.
- Rule-set definitions for automated determination of jurisdiction and applicable law according to a to-be-concluded international contract of an electronic value-added service.
- Prototype implementation and functional evaluation of the decision making engine.

This section is organized in two parts. Section 4.1 describes the work regarding the impact of economic objectives. It provides an overview of the business oriented approach for managing DiffServ networks, it presents the various business indicators used, and provides examples of the policy parameter derivation process. Section 4.2 addresses a prototype implementation of a rule-based system to determine applicable law according to the relevant EU Rome I regulation [10].

4.1 Business-driven Management of DiffServ Networks

Several intra- and inter-domain Quality of Service (QoS) provisioning mechanisms for IP networks have been researched and developed using the Differentiated Services (DiffServ) technology. However, the requirements and implications in managing DiffServ networks from a business-oriented viewpoint have received relatively little attention. QoS management has always been one of the most popular application domains of policies since ISPs can realize their objectives through flexible programmability with respect to

offered services and treatment of customer traffic in their network. Previous work in this area has shown how low-level configuration policies can be derived from service-level objectives via a refinement process. This work takes a step further by bridging the gap between business value and configuration management using a set of business indicators (BIs) that guide the refinement process.

In contrast to prior work that applied business-driven techniques to drive IT and SLA management for data centres [1][4], we aim here to tackle the issue of QoS management in DiffServ/MPLS networks. We describe an approach to control relevant business indicators and evaluate their relationships with service management objectives and policies. We focus on static and dynamic Admission Control (AC) of service subscriptions and extend previous work on policy refinement [16] by considering the influence of BIs when generating appropriate configuration policies. This is realized by defining a set of mapping functions that take into account the impact of BIs over service management policies. Administrator assigned weights of importance are given to each BI and are then used to derive appropriate policy parameters. We demonstrate, through simulation scenarios, the effectiveness and practicality of this approach, highlighting the issues necessary for its realization.

The work described in this section is based on a paper [15] that has been accepted for publication in IEEE/IFIP NOMS 2010, which is attached at the end of this report.

4.1.1 Business Indicators and Service Management Objectives

The QoS-based policy refinement process is based on a set of high-level objectives specific to service management functions. Such objectives aim, for example, to control the subscription volume, the service quality, and QoS degradation. These objectives can be achieved by policies that perform admission control and regulate service rates. The value of service management parameters set by policies however should reflect the business goals of ISPs, which are specified in terms of Business Indicators (BIs).

BIs can be used by service providers to define the management cycle of their QoS-oriented mechanisms, such that the negative impact of incidents affecting the business value is minimized. They relate to service management objectives (SMOs) and subsequently control the configuration parameters of derived policies. We have considered three types of BIs that can potentially be used by a service provider: *service satisfaction*, *profit*, and *loss*. The first relates to the level at which a service is perceived by the user – e.g. contractual rates, service quality and availability – whereas the last two relate to the revenue generated by contracted services. They are further discussed below.

Profit is directly proportional to the subscription volume, since more subscriptions imply higher revenue, but it is inversely proportional to quality because with lower quality levels (e.g. in terms of BW) more services can be accommodated in the available resources. Although more profit can be generated by over-subscribing, the risk of not delivering a service to the contracted quality level increases. This has a negative impact on the service satisfaction BI, which is directly proportional to the service quality.

The service satisfaction BI also relates to SMOs that control QoS degradation, new service invocations, and active services. High levels of satisfaction imply above average service rates for active services, high service availability in terms of invocation requests, and late QoS degradation preventative actions. Profit can be negatively influenced by penalties that an ISP has to pay when a service is not delivered to an acceptable standard, i.e. SLA violations. High losses can be incurred by very early (poor usage of resources) or delayed

(increased risk of congestion) QoS degradation prevention actions, and also by providing below contractual average service rates.

4.1.2 Deriving Policies from Business Indicators

It is evident that there are a number of trade-offs to be considered when achieving the various service management objectives. Based on the relationships described above, we propose an approach for deriving policies and associated parameter values to reflect the importance of the defined BIs. The approach aims at bridging the gap between business and QoS management and is based on a three-level hierarchy depicted in Figure 24. BIs are defined at the upper layer where each is assigned with a weight representing its importance in achieving a business-level objective. Weights take values from 0 to 1, the former signifying the least importance of a BI to an ISP.

The BIs are directly linked to individual SMOs which they influence. It is common for a BI to apply to more than one SMO. In this case the proposed approach assumes that the impact of that BI on the relevant SMOs is of the same magnitude, i.e. $W_2=W_{21}=W_{22}$. The impact of an indicator is represented with a set of mapping functions which take the associated weight into account. These are used during the refinement process to derive the value of policy parameters according to the business-level objective. In contrast to previous work on refinement that could only maximise or minimise a SMO, our approach provides a continuous range over which a parameter can be mapped and can thus achieve a more accurate configuration output.

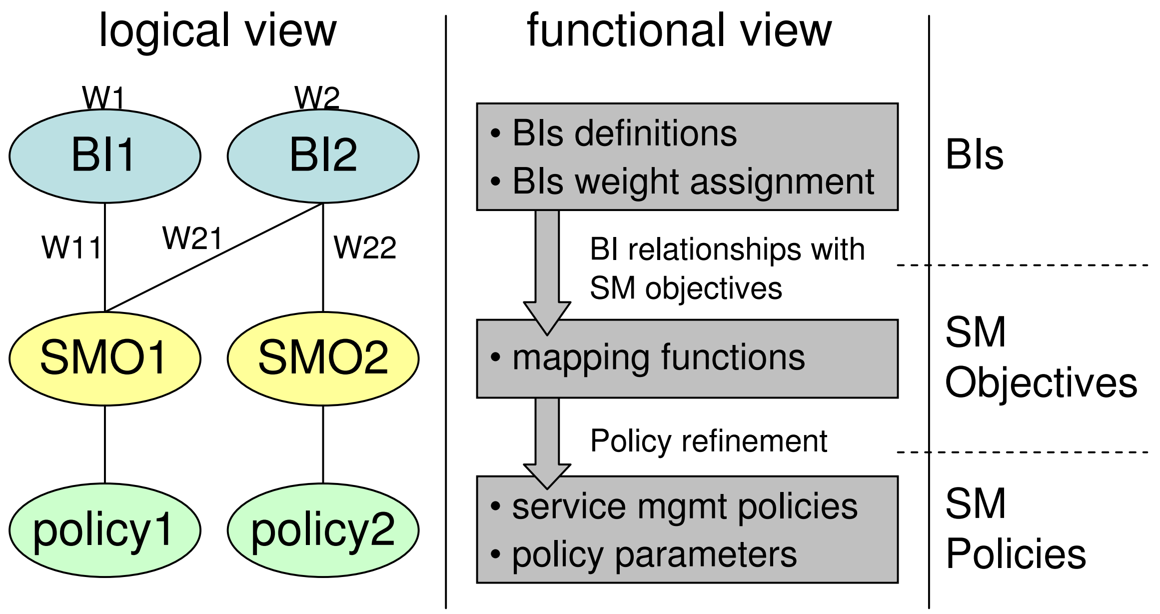


Figure 24: Relationships between BIs, Objectives, and Policies

The left part of Figure 24 shows that a SMO can be influenced by more than one BI. The approach treats the effect of those indicators independently by providing separate weighted linear mapping functions, e.g. for W_{11} and W_{21}. The resulting value of a policy parameter is a determined by averaging the output of individual functions.

4.1.3 SLS Subscription Mapping Functions

Based on the SMOs and the policies detailed in [15], this section gives examples of how policy parameter values can be derived from BIs. Profit and service satisfaction (*servSatisf*)

are the BIs influencing the management of SLS subscriptions (SLS-S). As shown in Figure 25, the policy setting a subscription upper (SU) limit – *setUpprLmt* – partly achieves

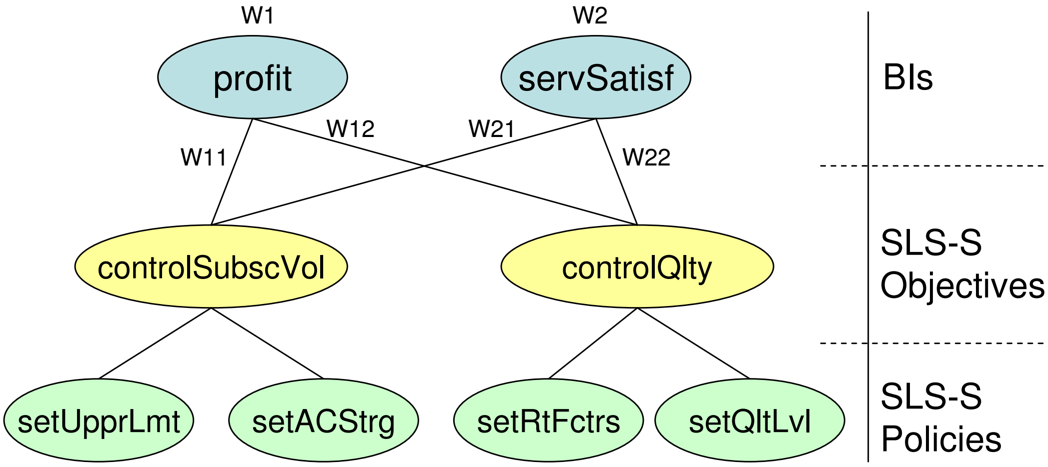


Figure 25: Relationships between SLS-S BIs, Objectives, and Policies

the *controlSubscVol* (control subscription volume) objective, which is influenced by both BIs. Setting the limit close to R_{max} (high end of a resource availability buffer) can maximise the subscription volume.

This can result in high profit but also in eventual network congestion – due to over-subscription – which can compromise service satisfaction. SU values close to R^{w}_{min} imply minimal risk of congestion but less profit due to lower subscription volumes. It is defined as:

$$SU = R^{w}_{min} + \frac{1 + W_{11} - W_{21}}{2}(R_{max} - R^{w}_{min})$$

The *controlSubscVol* objective is also achieved by the policy that sets the traffic demand (TD) parameter to be used in the static admission control decision - *setACStrg*. If TD_{max} is selected (conservative approach) profit suffers, since fewer subscriptions can be accepted, whereas higher satisfaction is achieved because there is confidence that services will be provided at their fully satisfied (FS) rates. The use of TD_{min} (less conservative approach) however, will result in higher profit, but lower satisfaction since there is confidence that services will only be provided at their almost satisfied (AS) rates. We define TD as follows:

$TD = TD_{max}$, when $W_{11} \le W_{21}$

$TD = TD_{min}$, when $W_{11} > W_{21}$

The two indicators also affect policies achieving the *controlQlty* objective, i.e. those setting service rate factors (range from 0 to 1) and quality levels (range from -1 to 1). AS factors close to 0 and FS factors close to 0.5 imply increased service satisfaction due to the higher rates produced, but can result in lower profit as fewer service subscriptions can be accommodated. Opposite results are achieved with ASs close to 0.5 and FSs close to 0 because of lower service rates. The final value of individual factors is determined by averaging the result of applying the weights of the BIs, e.g. $Fctr_{AS}$= ($Fctr_{AS1}$+$Fctr_{AS2}$)/2.

$$Fctr_{AS} = \frac{1}{4}(1 + W_{12} - W_{22})$$

$$Fctr_{FS} = \frac{1}{4}(1 - W_{12} + W_{22})$$

The second policy relating to the *controlQlty* objective sets the overall quality level (OQL). Values close to 1 suggest that services are provided at their FS rates, even during congestion. This results in higher satisfaction but lower profit since a less number of services can be accommodated within a given resource availability buffer. Satisfaction is reduced with OQL values close to 0 as services are most likely to be provided at their AS rates, but an increased profit is achieved with more accepted subscriptions. OQLs close to -1 result in services with no guarantees. Satisfaction is at the lowest possible level whereas profit is at the highest.

$$OQL = (W_{22} - W_{12})$$

Similar mapping functions have been defined for the dynamic admission control of services where the loss BI influences the derivation of policy parameters, details of which can be found in [15].

4.1.4 Evaluation and Achievements

The practicality and effectiveness of the approach has been evaluated based on an illustrative scenario in which business-related and DiffServ networking issues were put into context. In this scenario, we consider a network service provider that implements the business-driven approach, offering SLAs with bundles of FTP, web browsing and e-mail services. The provider classifies services into medium and high quality, assigning them to different DiffServ QoS classes. The evaluation was carried out with a set of experiments that were conducted with a modified OPNET toolkit, extended to support the execution of dynamic admission control policies. The influence of weighted BIs on the configuration of the network, as a result of dynamic traffic fluctuations, was analyzed demonstrating the benefits of the approach. Details of the experimental results can be found in the relevant paper [15] attached at the end of this document.

In conclusion, this work described the elements required to bridge the gap between business and configuration management in the DiffServ QoS management domain, elaborating on a bidirectional approach that eases the analysis of business-driven DiffServ management strategies. The approach relies on business indicators that are used by ISPs to derive a network configuration that is in line with high-level business objectives. We considered *service satisfaction*, *profit*, and *loss* BIs and their influence on both static and dynamic admission control of services. The approach was evaluated through simulation experiments.

4.2 Automated Determination of Jurisdiction and Applicable Law

The objective of the process presented here is to determine the country of jurisdiction or applicable law in case of disputes in the context of value-added services. The simple diagram depicted in Figure 26 illustrates the decision process, which, by taking a set of input parameters (decision rules, SLA specifications), it applies the necessary logic to determine the relevant country, *i.e.*, the output.

4.2.1 Definitions and Assumptions

Before defining the decision rule-set, it is necessary to identify the parameters that need to be considered when making a decision. These parameters need to be established during the formation of the contract and are instantiated in the SLA specifications. The following parameters listed have been determined in accordance with the EU Rome I regulation [10] as modeled in the paper "*Automated Contract Formation for Electronic Value-added*

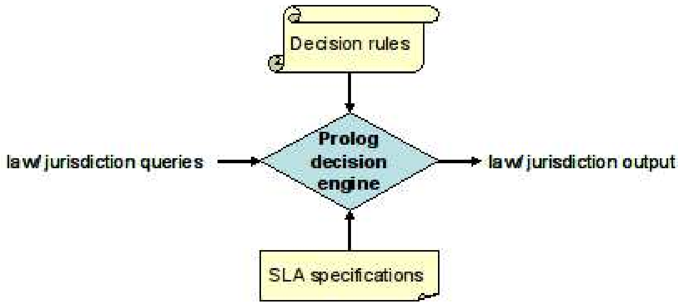


Figure 26: Decision Process Overview

Services in the Internet—The Case of Bandwidth-on-Demand Contracts in Europe" [29] published and presented at the 37th Research Conference on Communication, Information and Internet Policy (TPRC 2009). Figure 27 depicts the accordingly modeled UML activity diagram embracing actions and decisions required to determine applicable law for international contracts that fall under application of the Rome I regulation.

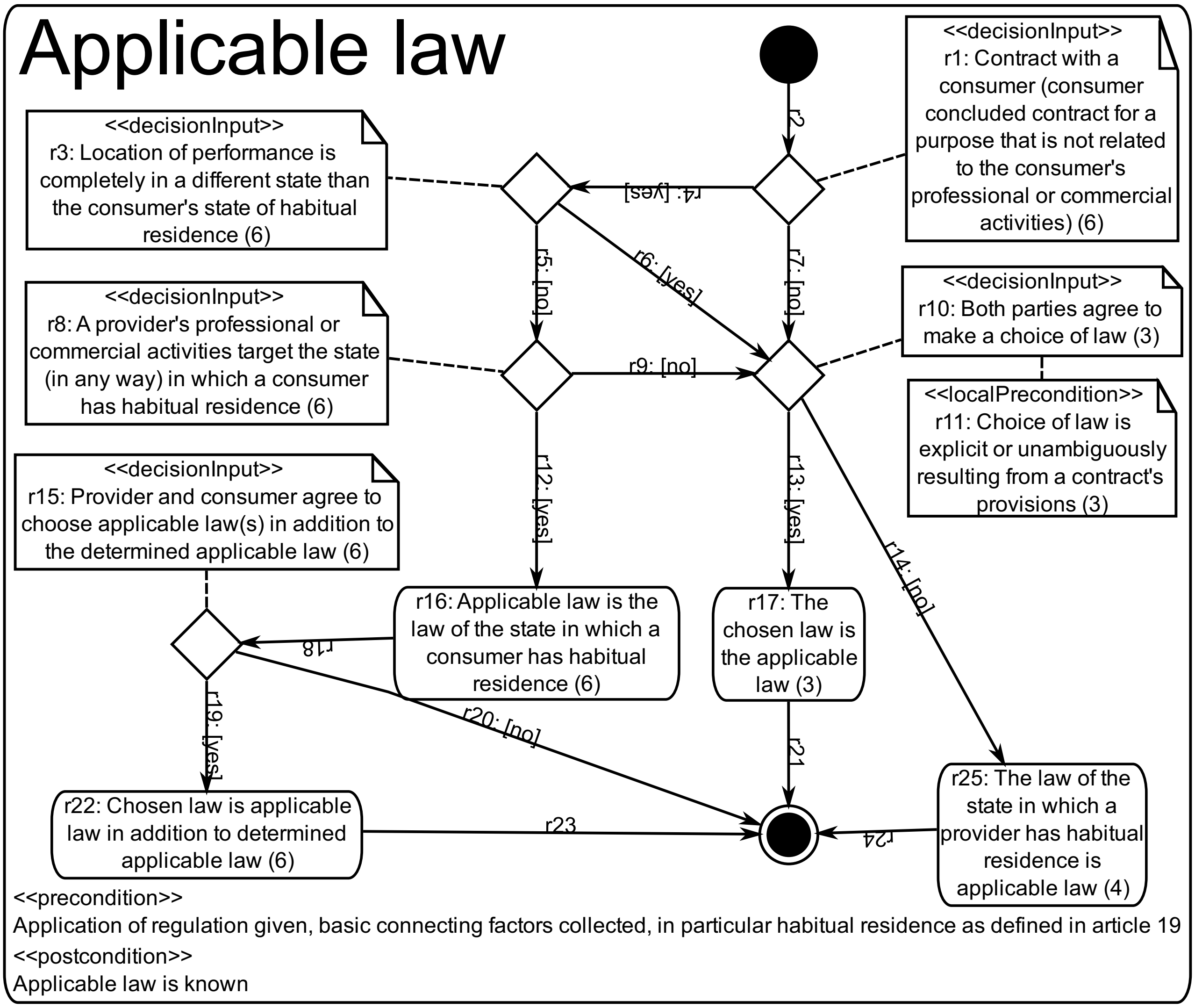


Figure 27: Activity Diagram (for Rome I Regulation [10]) to Determine Applicable Law [29]

4.2.1.1 Applicable Law Parameters on European Level

- *ConsType* This parameter takes one of two values – private or business. It represents the type of service consumer, which can be an individual or a business.
- *ConsHabRes* Specifies the country representing the habitual residence of the consumer.
- *ProvHabRes* Specifies the country representing the habitual residence of the provider.
- *PerfLoc* Specifies the country in which the location of performance is.
- *ProvProfComScope* Specifies the country in which the provider has professional or commercial activities.
- *ExpLawChoice* Specifies the country of which the law applies, if this has been explicitly agreed upon between the two contracting parties.

In addition to the above parameters, SLA specifications also include:

- *SLAid* Identifies an SLA.
- *ServcType* Identifies the type of service specified in an SLA.

Based on the above parameters, SLAs can be specified in the form of compound logic terms (sla) as follows:

```
sla(SLAid, ServcType, ConsType, ConsHabRes, ProvHabRes, PerfLoc,
  ProvProfComScope, ExpLawChoice)
```

4.2.1.2 Rule-set on European Level

The rules that decide the country of applicable law or jurisdiction can be represented in logic form where the head of the clause represents the goal to satisfy (in this case the *applLaw* and *juri* predicates), and its body represents a set of conditions that have to be met:

```
applLaw(SLAid, ServType, ApplbLaw, AddApplbLaw) :-
  sla(SLAid, ServcType, ConsType, ConsHabRes, ProvHabRes, PerfLoc,
        ProvProfComScope, ExpLawChoice),
  (condition1; condition2; condition3).

juri(SLAid, ServType, Jurisdn) :-
  sla(SLAid, ServcType, ConsType, ConsHabRes, ProvHabRes, PerfLoc,
        ProvProfComScope, ExpLawChoice),
  (condition4; condition5; condition6).
```

The arguments of the *sla* term are unified with ones in SLA instances in the Prolog knowledge base. These are subsequently used in comparisons in the conditional part of the rule. The various decisions in the process of deriving applicable law/jurisdiction are based on the conditions (or constraints) defined in the rule(s). The disjunction of terms in the conditional part signifies the various paths that the process can follow in the UML activity diagrams.

- *ApplblLaw* This is an output parameter, i.e. the primary country of applicable law.
- *AddApplblLaw* This is an output parameter, i.e. the secondary country of applicable law.
- *Jurisdn* This is an output parameter, i.e. the country of jurisdiction.

The above parameters are unified during the decision making process, which can be invoked with 2 types of queries:

```
(a) ?- applLaw(SLAid, ServType, ApplbLaw, AddApplbLaw.
(b) ?- juri(SLAid, ServType, Jurisdn).
```

4.2.2 Prototype Implementation for Applicable Law

The following code listing shows a partial rule-set expressed in Prolog. It reflects and refers to the decisions, transitions, and actions contained in the activity diagram depicted in Figure 27. Rule-sets are formed in the way introduced in Section 4.2.1.2 taking into consideration those parameters described in Section 4.2.1.1. The prototype implementation as was found fully functional as it produced for the complete set of test cases run an output with respect to applicable law which was fully in-line with the respective expected result when following the according path through the activity diagram of Figure 27.

```
applLaw(SLAid, ServType, ApplbLaw, AddApplbLaw):-
 sla(SLAid, ServType, ConsType, ConsHabRes, ProvHabRes, PerfLoc,
     ProvProfComScope, ExpLawChoice),

%result1
   % path1 (r1,r10)
 ((((ConsType\==prvt,
    ExpLawChoice==[]);  %if parties dont agree it means that the
 ExpLawChoice field is empty
   % path2 (r1,r3,r10)
   (ConsType==prvt,
    PerfLoc\==ConsHabRes,
    ExpLawChoice==[]);
   % path3 (r1,r3,r8,r10)
   (ConsType==prvt,
    PerfLoc==ConsHabRes,
    ProvProfComScope\==ConsHabRes,
    ExpLawChoice==[])),
  ApplbLaw=ProvHabRes);

%result2
   % path4 (r1,r10)
 (((ConsType\==prvt,
    ExpLawChoice\==[]);
   % path5 (r1,r3,r10)
   (ConsType==prvt,
    PerfLoc\==ConsHabRes,
    ExpLawChoice\==[]);
   % path6 (r1,r3,r8,r10)
   (ConsType==prvt,
    PerfLoc==ConsHabRes,
    ProvProfComScope\==ConsHabRes,
    ExpLawChoice\==[])),
  ApplbLaw=ExpLawChoice);

%result3
   % path7 (r1,r3,r8,r15)
 (...);

%result4
   % path8 (r1,r3,r8,r15)
 (...).
```

4.2.3 Preliminary Achievements

IELOS has achieved a number of highly promising results as follows:

- From a methodological point of view, a viable and fully functional way was designed and implemented to determine in a legally compliant, automated manner the respective applicable law or jurisdiction according to a previously modeled existing law. The method to model a law as a UML activity diagram is preliminarily documented in a successfully published paper [29], while the method to implement such a modeled law in Prolog was for the first time achieved and preliminarily documented in this deliverable.
- The underlying set of parameters needed for the implementation was initially laid out. Since for ongoing and future work, the scope of modeled laws and implemented laws shall be extended, the need for a consolidated information model has become apparent. IELOS and SLAP'N (see Section 5) profit from an established fruitful mutual result exchange in the way that SLAP'N focuses on the development of a solid information model for determining jurisdiction and applicable law for international service contracts, whereas IELOS concentrates on the respective functional aspects facilitating an implementation of the related procedures.
- Ongoing work in IELOS is concerned with developing and adapting a re-fined implementation technique which reflects paths traversed in a considered UML activity diagram in a more structured way. This new technique results in a code basis that allows for higher modularization and improved functionality assessment according to defined use cases. Moreover, this technique produces better readable code with shorter rule statements, plus it incorporates naming schemes which are consistent with the information model determined in SLAP'N. Finally, the so far considered case of applicable law is currently extended by the respective case of jurisdiction determination.

5 SLAP'N — SLA Planning and Negotiation

SLAP'N aims to develop an information model in the field of SLA management (SLM) that supports the planning and negotiation of service contracts with international connection. International connection in this context may relate to contract parties that reside in different legal domains and/or it may relate to an electronic service that is to be provisioned between multiple legal domains. This SLAP'N information model shall extend the well established information model for IT service management (ITSM) as developed and addressed in parts in previous EMANICS WP8 SLM work [30][31] and as fully documented findings and outcomes of the PhD thesis entitled *IT-supported Service Level Management - requirements and specifications for a management architecture* [24], supported by EMANICS.

SLAs constitute a key element in service contracts which, in turn, embrace the set of relevant contractual obligations to be fulfilled and negotiated—negotiations that are of particular interest in many use cases for which future Internet management mechanisms and techniques will be applied. While previous stages of WP8's joint research activities in the context of SLAs and SLM were focused on developing a formal service and SLA model (including static aspects as well as service dynamics), creating a common understanding of the deployment scenarios for SLAs (where can/should they be used?) and providing guidance in mapping Quality of Service (QoS) and Quality of Device (QoD) metrics, the process of planning and negotiating SLAs from a service lifecycle perspective has not been addressed by EMANICS WP8 yet. Thus, SLAP'N targets for the completion of the overall Economic Management Model.

Furthermore, SLAs were, so far, considered to be negotiated and concluded between a service provider and service customer within a single administrative domain—typically resulting in a situation in that both parties are also located in the same legal domain, such as in a single nation. In the context of EMANICS, which represents a European network of excellence, this single-domain perspective shall be extended to fully support multi-domain cases. Hence, the impact on SLA—in more general terms on service contract—negotiation and conclusion for international (contract) relations is investigated. In particular, the automated determination of jurisdiction and the law applicable to a contract is looked at in further detail. Jurisdiction and applicable law constitute two core elements of every concluded contract, and they gain even more importance in international contracts. To that aim, necessary adaptations and potentially required extensions to the previously developed information model are envisaged within the frame of SLAP'N.

5.1 Objectives and Scope

Figure 28 shows the initial starting point for SLAP'N with respect to the applicable business relationship considered. This relationship covers a service provider and a service customer, whereas it has to be noted that a service customer may or may not be the same enitity as the service user. Within the scope of SLAP'N, thus, a differentiation between that party that uses and that pays for a service is made. Figure 28 also includes the object of such a provider/customer business relationship, namely the contracted service to be provided and consumed, respectively.

According to ITIL [23], a service can be regarded as a means of delivering value to customers by facilitating outcomes customers want to achieve without the ownership of specific costs and risks. As a specialization of this definition, an electronic service can be defined as a service that can be realized exclusively by means of electronic systems and

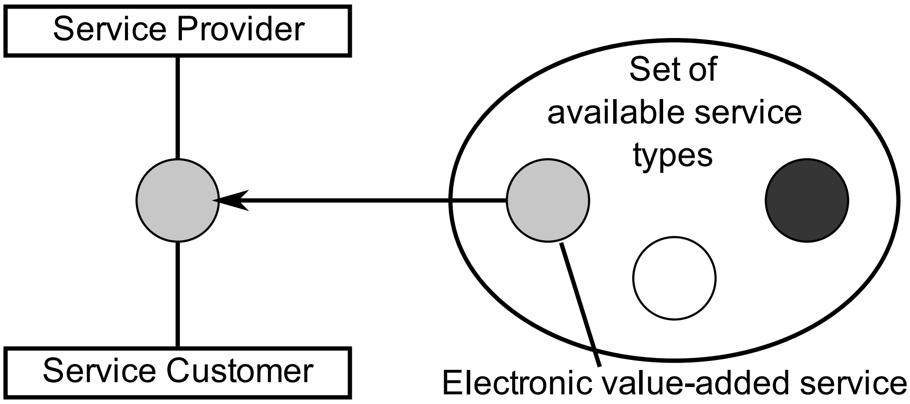


Figure 28: Provider/Customer Relationship

information technology equipment as well as through aggregation and interconnection of such systems. In SLAP'N, electronic value-added services are focused exclusively.

Figure 29 develops the previously introduced customer/provider business relationship further by including potential internal or external suppliers—parties which are possibly not known to a service customer—and potential service users and further service customers (when re-selling a service)—parties possibly not known to a service provider. It is important

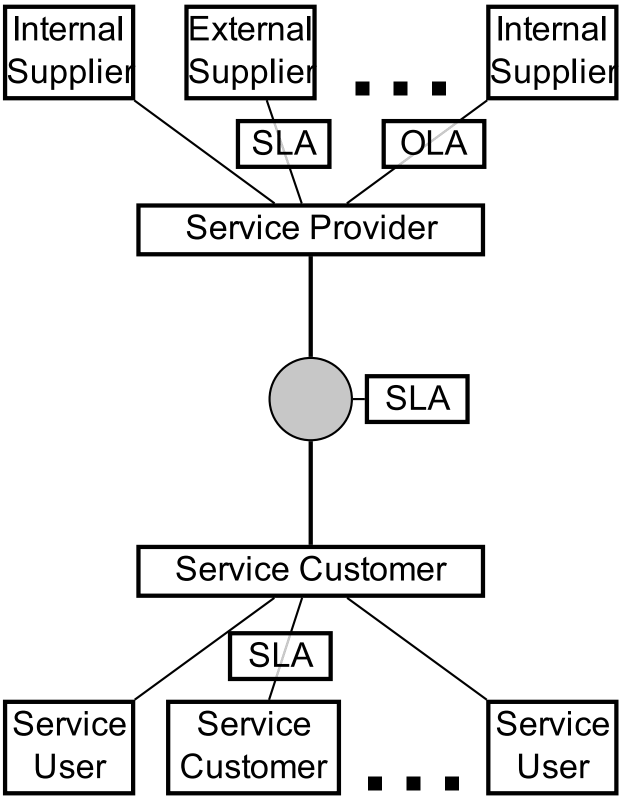


Figure 29: Extended Provider/Customer Relationship

to note that SLAP'N focuses on the primary business relationship between (exactly) one service customer and (exactly) one service provider. SLAP'N, thus, abstracts from potential further business relations within the scope of a service provider or a service customer, respectively. This implies that SLAP'N envisages a bilateral contractual agreement for an electronic value-added service as negotiated between the respectively involved service provider and service customer only. Such a contractual agreement may find a technically

measurable representation in an associated SLA. SLAP'N is concerned with those concepts and information artefacts that allow the legally compliant formation of an international service contract. In particular, concepts and artefacts are focused that help determining jurisdiction and applicable law.

Jurisdiction and applicable law constitute two key contractual provisions in any international contract of civil and commercial matters. While jurisdiction indicates which nation's courts are authorized to hear and decide on a potential conflict arising from a concluded contract, applicable law indicates under which nation's law a court decision shall be found. The legal frame to be consistent with when determining jurisdiction and applicable law for an international contract of civil and commercial matters is laid out by the respective (national and supra-national) provisions of Private International Law (PIL). PIL dictates to relate jurisdiction and applicable law to a nation according to the contract-specific set of relevant connecting factors. These connecting factors embrace facts which originate, on the one hand, from the respectively involved contract parties and, on the other hand, from the resulting contractual obligations.

For commercially provided electronic services in the Internet, the (legally compliant) determination of jurisdiction and applicable law, and with that the set of connecting factors, is of key importance to both, a service provider and a service customer. First, commercial service provision asks for a contract to be concluded between service provider and service customer in the first place. Second, with the Internet being a global infrastructure, such a contract often will be an international contract. Thus, jurisdiction and applicable law are of importance, in principle. Third, a contract has to be concluded before the actual service is provided. Accordingly, jurisdiction and applicable law need to be determined prior to service provisioning as well. Fourth, a contract of electronic service provisioning in the Internet falls under the category of a contract of civil and commercial matters. This implies that the respectively applicable PIL imposes, in principle, the procedures to determine jurisdiction and applicable law in relation to international contracts of electronic service provisioning in the Internet. Accordingly, the relevant set of connecting factors needs to be identified and used.

In summary, SLAP'N addresses the following specific objectives within the determined work scope:

- Formal modeling of a bilateral **service provider/service customer relation** for an electronic **value-added service**.
- Development and documentation of an **information model** which covers the complete set of required concepts and information artefacts to determine **jurisdiction** and **applicable law** for the provider/customer relation envisaged. This bases on the fundamental assumption that service provider and service customer are not located within the same legal domain and/or that a service needs to be provided between different legal domains, so that an **international contract** is to be concluded.

5.2 Methodology

In consideration of the scope outlined, the SLAP'N methodology has been determined as follows. SLAP'N considers three main inputs and two main outputs. The first input reflects the existing ITSM information model [24]. This information model constitutes both, a comprehensive viewpoint on ITSM and, by that, the well suited initial starting point for potential extensions and adaptations needed in the course of SLAP'N. Section 5.3 provides an overview of the respective concept model and the changes needed within the scope of

SLAP'N. In order to assess whether and which extensions or adaptations to this ITSM information model are needed, the second and third main inputs to SLAP'N determine a set of functional requirements.

The first set of requirements originates from an extensive contract qualification of a complex international service contract construct as researched and documented in [29]. These requirements help develop the appropriate SLAP'N concept model. The second set of requirements originates from experience gained throughout the EMANICS WP8 project IELOS as described in Section 4.2. These requirements help develop the appropriate SLAP'N information artefacts. IELOS and SLAP'N, thus, profit from a fruitful mutual exchange of research results in the sense that SLAP'N focuses on a solid information model for use when determining jurisdiction and applicable law for international service contracts, while IELOS facilitates an implementation of those jurisdiction- and applicable law-oriented workflows modeled. In other terms, SLAP'N supports an automated and legally-compliant determination of jurisdiction and applicable law by means of an appropriate information model, whereas IELOS brings in support in terms of functional and implementation aspects.

As with respect to those two main SLAP'N outputs mentioned, the first output consists in a concept model, while the second consists in the appropriately modeled information artefacts. This method is in-line with the information modeling approach adopted in [24]. The SLAP'N concept model aims to refer to its roots, *i.e.*, to the existing ITSM concept model, by explicitly mentioning points of intersection between the domains of SLM and international contracting. This includes that differing viewpoints in concept modeling shall be outlined and depicted. Since the SLAP'N concept model, however, tries to clearly separate service management and contact management domains, the set of existing concepts (of the service management domain) is foreseen to require adaptations in details only to support the PIL-compliant determination of jurisdiction and applicable law. Accordingly, changes are expected ex ante to consist mainly in concept model extensions rather than in concept model modifications. Furthermore, the SLAP'N concept and information artefact modeling envisages a model granularity that reflects, in a first step, a minimum set of concepts and artefacts required. This approach implies that there may be future concept and artefact extensions needed to the information model determined in this context.

5.3 SLAP'N Information Model in Detail

As stated above, the architecture provided in this section encompasses informational aspects, while functional aspects find representation by means of requirements derived from insight gained in those implementation-driven efforts conducted in the course of IELOS (*cf*. Section 4.2). Thus, the SLAP'N model consists of a formal information model covering international contracting- and service management-related information objects and using UML class definitions/diagrams as a modeling language.

5.3.1 Information Modeling Guidelines

The following questions indicate goals of an information model for SLM, such as the one developed in [24] which serves as the starting point for the SLAP'N information model:

- Which information needs to be processed and maintained in the context of SLM and its activities?

- Which information object (artifact) definitions can be used to bundle all relevant information in a clear set of object classes?
- What are the concrete informational requirements with respect to the identified information objects?
- What are the superior interrelations, dependencies and multiplicities between these artifacts?

Once these questions have been answered, a refinement of the information model into concrete data models needs to be performed, addressing the following points:

- How can the informational requirements of the artifacts be refined and formalized as data models?
- What are the concrete data models resulting from the set of information object classes identified before?
- How can all data models be integrated into one single consistent information system for SLM?

Those questions raised and listed constitute not only the relevant information modeling guidelines [24], but also in SLAP'N. While SLM was the primary focus in [24], these questions need to be slightly adapted for the scope of SLAP'N. SLM still plays an important role, but the respective concepts and information objects facilitating a PIL-conformant determination of jurisdiction and/or applicable law have to be added. In other terms, this means that the SLAP'N information model needs concepts for international contracting, which, in turn, asks for specific artefacts to represent connecting factors such as a service customer's domicile or a contracted service's location of performance. Consequently, Section 5.3.2 conducts the required modeling step from the existing SLM-driven concept model to the extended SLAP'N concept model, whereas Section 5.3.3 documents the respectively modeled information artefacts.

5.3.2 Concept Model

Figure 30 shows the major concepts relevant to the existing ITSM model as documented in full detail in [24]. This model serves as a starting point to determine the appropriate SLAP'N concept model. The existing model bases on an organization model [24] that covers the relevant set of organizational domains and actors from an SLM point of view. Main organizational domains to be considered embrace a service provider domain, a service customer domain, and external supplier domain. For each of these domains mentioned, a number of roles to be adopted by an actor is presented. When an actor plays a given role, that actor can be either responsible (operationally responsible), accountable, informed, supportive, consulted, or verifying.

In consideration of the concept model depicted in Figure 30 and these indications on the respective underlying organizational model, the main questions to be addressed here is whether there are any changes needed for the SLAP'N concept model and, if yes, which changes in the existing concept model there need to be. In order to answer these questions, a comprehensive case study on the complex service contract construct of a virtual private layer 2 circuit with a dynamic bandwidth feature [29] is considered. This case study, which investigates a product offered on the market, assesses all relevant contract elements, derives the according contract obligations both contract parties have to maintain, and it simulates a contract qualification—aiming to determine the respective applicable contract nature and, with that, the suited contract type. Subsequently, insight gained from

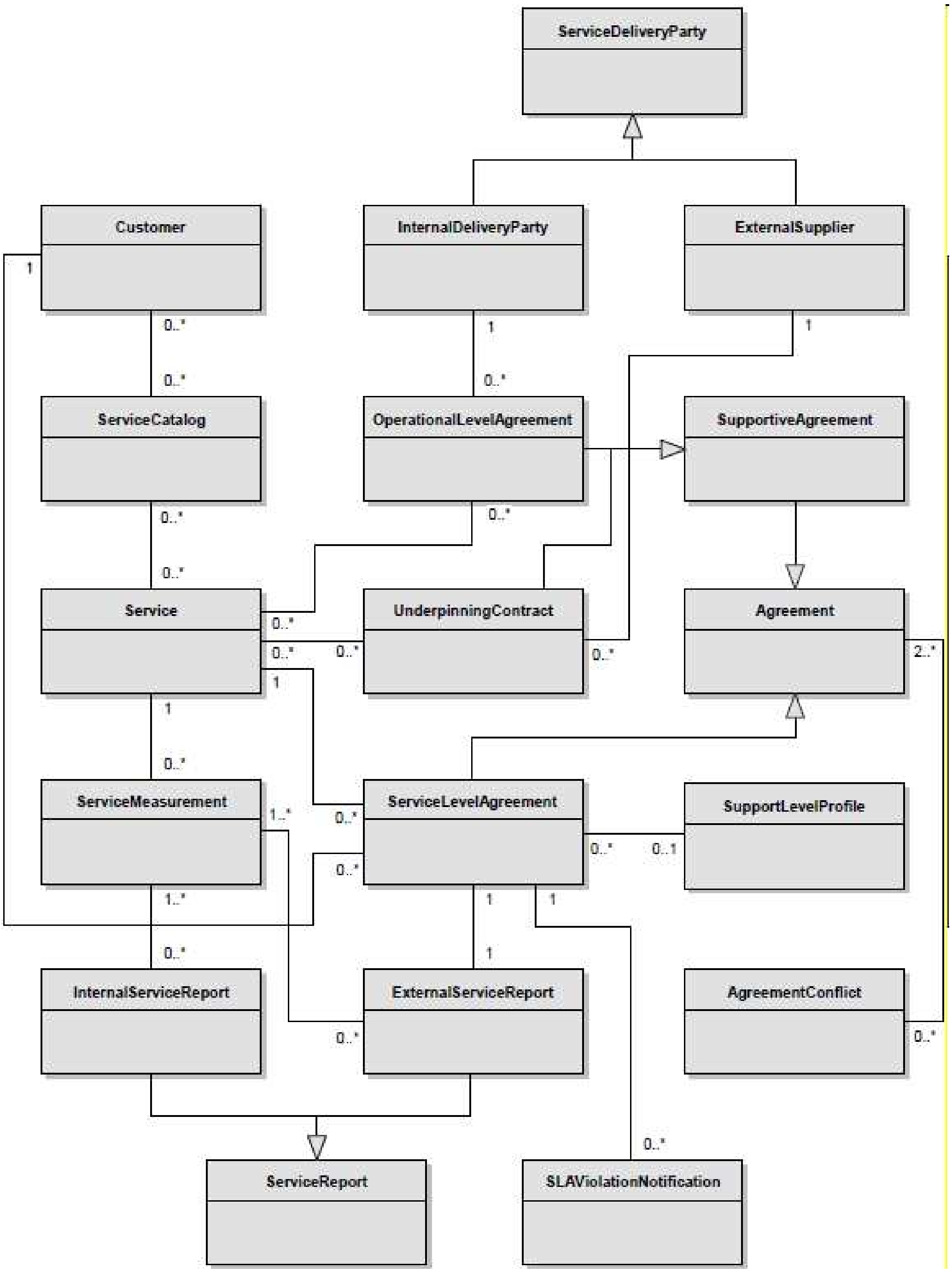


Figure 30: Partial View on Existing Concept Model [24]

this case study and the SLAP'N scope outlined (*cf*. Section 5.1) are investigated to determine requirements for the SLAP'N concept model.

The concept model shown in Figure 30 shows relevant concepts and their interrelations from a service provider perspective. This perspective is appropriate for the existing model, since that model adopts an SLM point of view. The management of an already contracted service lies primarily within the domain of the service provider. Accordingly, the service provider is not reflected by means of a specific concept. Consequently, only concepts for those organizational domains with which a service provider interacts—again, from an SLM point of view—are included in Figure 30.

For the SLAP'N concept model, however, this focus on SLM-driven concepts requires extension towards an angle covering both, service management and contracting concepts. Service management-driven concepts are relevant to SLAP'N as well, since service contracts are looked at. Thus, contracting issues and provisioning/management aspects of contracted services are both of interest. As a contract is a mutual agreement, the inclusion of contracting concepts requires to reflect the respective involved contract parties equally. In accordance with SLAP'N's scope definition, considered contract parties embrace exactly one service provider and exactly one service customer (both being of type ContractParty). On the other hand, concepts of type ServiceDeliveryParty are not focused on within SLAP'N so that these concepts can be neglected—which, however, does not imply that these concepts are not relevant. They are seen unchanged with respect to the existing model, they are only not included in the SLAP'N concept model due to an inherently adapted perspective in the SLAP'N concept model. Consequently, a first set of adaptations for the SLAP'N concept model is summarized as follows:

Adaptation 1: Concepts for both considered contracting parties included. Existing Customer concept renamed to ServiceCustomer, complemented by its counter-part concept, ServiceProvider. ServiceCustomer and ServiceProvider inherit from (newly included) concept ContractParty.

Adaptation 2: ServiceDeliveryParty concepts (including concretized concepts of InternalDeliveryParty, ExternalSupplier) abstracted away from concept model.

In direct consequence of including contracting concepts, while keeping (key) provisioning an management-related issues focused, those two originally centered concepts of Service and ServiceLevelAgreement have to be completed by the respective centered concept from a contracting perspective, the ServiceContract concept. The existing concept model knows contractual concepts as well. These include on the one hand concepts representing specializations of the concept Agreement, namely ServiceLevelAgreement, OperationalLevelAgreement, and SupportiveAgreement. On the other hand, they include concepts for the Agreement-associated AgreementConflict concept as well as the concept of UnderpinningContract. With the exception for ServiceLevelAgreement, these concepts listed are abstracted away from the SLAP'N concept model. With respect to OperationalLevelAgreement and UnderpinningContract, i.e., the two concepts of type SupportiveAgreement, this happens as a consequence of Adaptation 2. Agreement and the associated AgreementConflict concepts are masked in order not to confuse concepts that are newly included by introducing the ServiceContract concept. Before explaining adaptations required by ServiceContract, a second set of adaptations for the SLAP'N concept model is summarized as follows:

Adaptation 3: Concept for ServiceContract as a central contracting concept included.

Adaptation 4: SupportiveAgreement concepts (including concretized concepts of OperationalLevelAgreement, UnderpinningContract) abstracted away from concept model.

Adaptation 5: Agreement (including associated concept of AgreementConflict) abstracted from concept model due to potentially misleading interpretation of ServiceContract concept.

ServiceContract, ServiceLevelAgreement, and (the abstracted) Agreement concepts are content-wise closely related and located within the outlined scope of SLAP'N. Nevertheless, these concepts have to be clearly differentiated. While, from an SLM perspective, an SLA constitutes the primary contractual element—an agreement—between

a service provider and a service customer, an SLA is usually a single contract part only when considering the full contractual agreement concluded between a service provider and a service customer.

The performed case study [29] shows a typical example of such a complete contract construct. It is important to note that such a contract construct consists of multiple contract parts. There are generally applicable contract parts and there are parts that are specific to a service which may be covered by the overall service contract. There may be one or several services (including all relevant service-specific contract parts) to be considered within the frame of a single service contract. These different contract parts are addressed part by part later on. At this stage of concept model development, though, the primarily relevant consequence out of the case study investigated is that ServiceLevelAgreement probably remains the central service contract part from an SLM perspective, since an SLA reflects those contract elements that may be associated with accountable units, whereas—from a contracting perspective—an SLA obtains less weight as it is a single contract part in a row of other, equally important contract parts, all of which are covered by a single service contract.

This implies two things. First, ServiceContract and ServiceLevelAgreement may be both of type Agreement. In order to not emphasize, however, that what a service customer and service provider conclude is a (complete) service contract and not an SLA alone, Agreement is masked (see Adaptation 5). Second, the relation between contract and SLA is as such that a service contract shall cover at least one contracted service which might or might not have an SLA attributed. An SLA cannot exist for itself, meaning it does not constitute a contract of its own. It constitutes a contract part. Due to an SLA's acknowledged importance from an SLM point of view, however, the concept of ServiceLevelAgreement is perceived to feature a dual contracting/management characteristic. Consequently, the SLAP'N concept model as shown in Figure 31 introduces the respective domains of contract management and service management. Each concept is placed in either the contract management domain or the service management domain—the only exception being ServiceLevelAgreement which is placed at the edge of both domains in order to emphasize its dual characteristic as described.

Analogously, the SLAP'N concept model emphasizes for each concept covered whether a concept is more closely related to the customer or the provider domain, respectively. To that aim, a concept is placed in either domain. Concepts, in which both contract parties have an equally important stake, are placed at the edge of both domains. This includes on the one hand all contract parts, since a contract is by definition a mutual, ideally balanced, agreement. On the other hand, the concepts Service and ServiceCatalog are placed at the edge of both, customer and provider domains, to clarify that the SLAP'N concept model that gives equal weight to both contract parties—one of which using and paying for a service, one of which providing and managing a service, both of which having contractual obligations of equal weight to fulfil. In that sense, a third set of adaptations for the SLAP'N concept model is summarized as follows:

Adaptation 6: Contract management and service management domains included. Concepts placed according to relation on either one or both domains.

Adaptation 7: Customer and provider domains included. Concepts placed according to relation on either one or both domains.

As previously stated, some contract parts are general, while others are service-dependant. A typical example for generally applicable—service-independent—contact elements are a

frame contract (to be subsumed under the concept of ServiceContract) and general terms and conditions (to be subsumed under the newly introduced GeneralTermsAndConditions concept). A service contract might see none or several contract elements that may fall under the concept GeneralTermsAndConditions, but typically either none or exactly one general terms and conditions contract part is assumed.

In the same way that terms and conditions may apply to a service contract as a whole, terms and conditions may apply to a specific service that might be covered by a service contract. Thus, a (service-specific) concept of TermsAndConditions is introduced and associated with the Service concept. A service may have zero, exactly one, or several terms and conditions documents attached. Both, general terms and conditions as well as service-specific terms and conditions are placed in the contract management domains. Terms and conditions constitute important contracting instruments, whereas they show only little direct impact in the service management domain—at least as long as there is no technical metric to be observed which would originate from a terms and conditions document.

Additional typical contract elements with a service dependency comprise acceptable use policies (none, a single, or multiple policies per service) and SLAs. While SLAs and the respective concept were previously discussed content-wise, the placement of ServiceLevelAgreement is a special one. ServiceLevelAgreement is placed at the edges of contract management and service management domains as well as at the edges of customer and provider domains. The first is due to its dual characteristic as described, the second is due to an SLA's nature of a contract element. ServiceLevelAgreement, thus, sees a fourfold characteristic with respect to related domains. With these service-specific and contract-general concepts outlined, a fourth set of adaptations for the SLAP'N concept model is summarized as follows:

Adaptation 8: Typical contract-general concepts, such as GeneralTermsAndConditions and (the previously introduced) ServiceContract, included. These concepts usually see a stronger relationship with the contract management domain than with the service management domain.

Adaptation 9: Typical service-specific concepts, such as TermsAndConditions, AcceptableUsePolicy, and ServiceLevelAgreement, introduced.

The existing SLM-driven concept model as shown in Figure 30 covers a concept Customer and it implies a provider perspective. With the introduction of a contract management and a service management domain, the need for a differentiated approach to contracting party and service using/provisioning parties becomes apparent. This is why, the two concepts of ServiceCustomer and ServiceProvider—both perceived as concepts relevant to the contract management domain—need to see a direct counterpart in the service management domain. From a management point of view, it is less important who signed and pays for a contract (ServiceCustomer), it is also less important who counter-signed a contract (ServiceProvider), while it is of key importance who consumes a service, thus, who uses a service, and consequently, who makes the service available. Accordingly, the contract-driven concept of a ServiceCustomer is complemented by a management-driven concept of a ServiceUser. And the contract-driven concept of a ServiceProvider is complemented by a management-driven concept of a ServiceOperator. In some cases, ServiceCustomer and ServiceUser may represent a single physical entity, namely a single natural person—the same might be true analogously on the provider/operator side for a specific contract/service—, but these two concepts are kept separate here to emphasize a

differentiated notion with respect to the respective domain a concept belongs to. In this sense, a fifth set of adaptations for the SLAP'N concept model is summarized as follows:

Adaptation 10: Management-driven concepts of ServiceUser and ServiceOperator as complement to their respective contract-driven concepts of ServiceCustomer and ServiceProvider included.

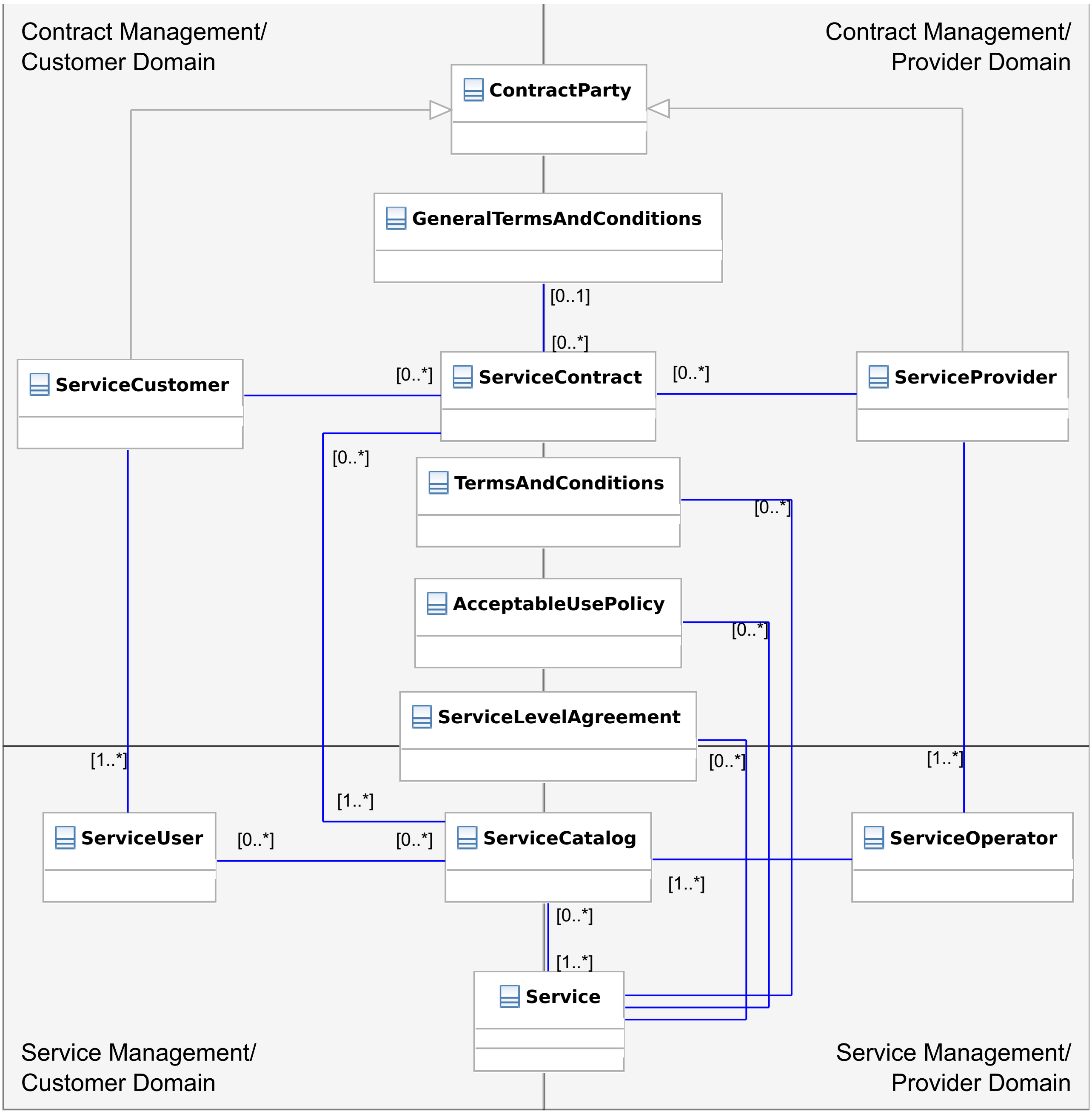


Figure 31: SLAP'N Concept Model

Finally, a number of ServiceLevelAgreement-related concepts of the existing model is abstracted from the SLAP'N concept model, whereas the two closely related concepts of Service and ServiceCatalog are transferred to the SLAP'N concept model. The set of masked concepts embraces ServiceReport (including concretizing concepts of InternalServiceReport and ExternalServiceReport), ServiceMeasurement, SLAViolationNotification, and ServiceLevelProfile. All these concepts are abstracted for a similar reason, which is that these concepts are primarily management-relevant. In a similar

way that, *e.g.*, ServiceDeliveryParty-related concepts were masked previously in the SLAP'N concept model, these SLA-oriented concepts are not explicitly mentioned in the SLAP'N concept model, but they are neither excluded completely.

In contrast, Service and ServiceCatalog are prominently referred to in the SLAP'N concept model due to their importance from a contractual and a management perspective. Despite being placed as concepts of the service management domains, services and service catalogs determine by definition key objects of any service contract. A service contract is understood to cover a service catalog, which reflects a customer-specific instance of a set of contracted single services. Accordingly, the concept of a ServiceContract is associated with the concept of a ServiceCatalog. ServiceCatalog, in turn, refers to services (and with that to the Service concept), while services see service-specific contract elements. A service contract is assumed to refer to at least one service catalog. A service catalog, analogously, is assumed to include at least one service.

In conclusion and completing the explanations on how to find from the SLM-driven concept model depicted in Figure 30 to the SLAP'N concept model depicted in Figure 31, a sixth set of adaptations for the SLAP'N concept model is summarized as follows:

Adaptation 11: ServiceReport concepts (including concretized concepts of InternalServiceReport, ExternalServiceReport) abstracted away from concept model. Accordingly, ServiceMeasurement concept (basis to prepare ServiceReport) abstracted away from concept model.

Adaptation 12: ServiceLevelAgreement-oriented concepts with a primary management focus—SLAViolationNotification and SupportLevelProfile—abstracted away from concept model.

Adaptation 13: Management-driven concepts of Service and ServiceCatalog used from existing model. ServiceCatalog associated with ServiceContract concept. Service concept associated with service-specific contract part concepts.

5.3.3 Information Artefacts

Driven by the SLAP'N concept model introduced, this section is concerned with the modeling of concrete, related information objects—information artefacts—required for a PIL-conformant determination of jurisdiction and applicable law. These artefacts, thus, reflect connecting factors which, in accordance with a specific modeled PIL, serve as main input to produce a list of recommendable jurisdictions and applicable laws. Table 9 documents the respective list of connecting factors [29] to be know from a European PIL point of view. Jurisdiction-relevant factors are summarized in Table 9 as derived for the Brussels I regulation [8], while applicable law-relevant factors originate from the Rome I regulation [10]. These connecting factors embrace those information objects that the preliminary implementation as developed within the frame of IELOS and documented in Section 4.2 uses for its primary in- and outputs.

Figure 32 shows the accordingly determined SLAP'N information model. Figure 32 represents a partial model which focuses on information artefacts in relation to the concept of a contract party, *i.e.*, in relation to a service customer and a service provider. The respective set of information artefacts needed to determine jurisdiction and applicable law for a service contract to be formed is twofold. First, and besides an object to uniquely identify a contract party, a number of objects are needed to characterize a service provider or a service customer. These artefacts cover location-oriented connecting factors (as

Table 9: Set of Relevant Connecting Factors in Brussels I and Rome I Regulations [29]

Dimension	Connecting factor	Remark
Contract party	Domicile	Location (state is sufficient)
	Establishment (s)	Location (state is sufficient)
	Habitual residence	Location (state is sufficient)
Contract	Contract type	Only service contracts considered here
	Performance	Location (state is sufficient)
	Choice of jurisdiction	Boolean, either “yes” or “no”
	Chosen jurisdiction	Location (state sufficient, representing court or courts of a state), only relevant if choice of jurisdiction made (“yes”)
	Exclusive choice of jurisdiction	Boolean, either “yes” or “no”, only relevant if choice of jurisdiction made (“yes”)
	Choice of law	Boolean, either “yes” or “no”
	Chosen law	Location (state sufficient, representing law or laws of a state), only relevant if choice of law made (“yes”)
	Consumer contract	Boolean, either “yes” or “no” (implies B2C service provisioning if “yes”, B2B provisioning if “no”)
	Provider target markets	Location (state is sufficient), only relevant in case of consumer contract (“yes”)
Dispute	Defendant	Contract party (implies that the respective other contract party in a bilateral contract is claimant)
	Dispute matter	Only disputes in relation to either a contract/contractual claim or operation of an establishment considered here
Result	Jurisdiction(s)	Location (state sufficient, representing court or courts of a state)
	Applicable law(s)	Location (state sufficient, representing law or laws of a state)

contained in Table 9 under the dimension of contract parties) such as a contract party's location of domicile, establishment (or establishments), and habitual residence. Furthermore, a contract party is characterized by an object reflecting a party's business role in a considered contractual relation, namely whether a party is provider or customer. In case, a provider is assumed to represent a professional service supplier of services, while a customer is assumed to represent either a private, non-professional (B2C, Business-to-Consumer) or a professional buyer (B2B, Business-to-Business), a customer may or may not be a consumer. To express this, the respective artefact Consumer is included in Figure 32.

Moreover, a second dimension to characterize a contract party is reflected in Figure 32 by a number of artefacts representing a party's (potential) wish to make a choice of jurisdiction and/or a choice of law. If a contract party envisages a choice, its preferred jurisdictions and laws, respectively, are modeled in the respectively included artefacts. In case of a choice of jurisdiction, this choice can be termed exclusive or non-exclusive. It is important to note that these artefacts mentioned in relation to a contract party are reflected Table 9 as well—in Table 9, however, these factors are attributed to a (service) contract. Figure 32, as a partial information model emphasizing on contract parties, provides an elaborate overview of

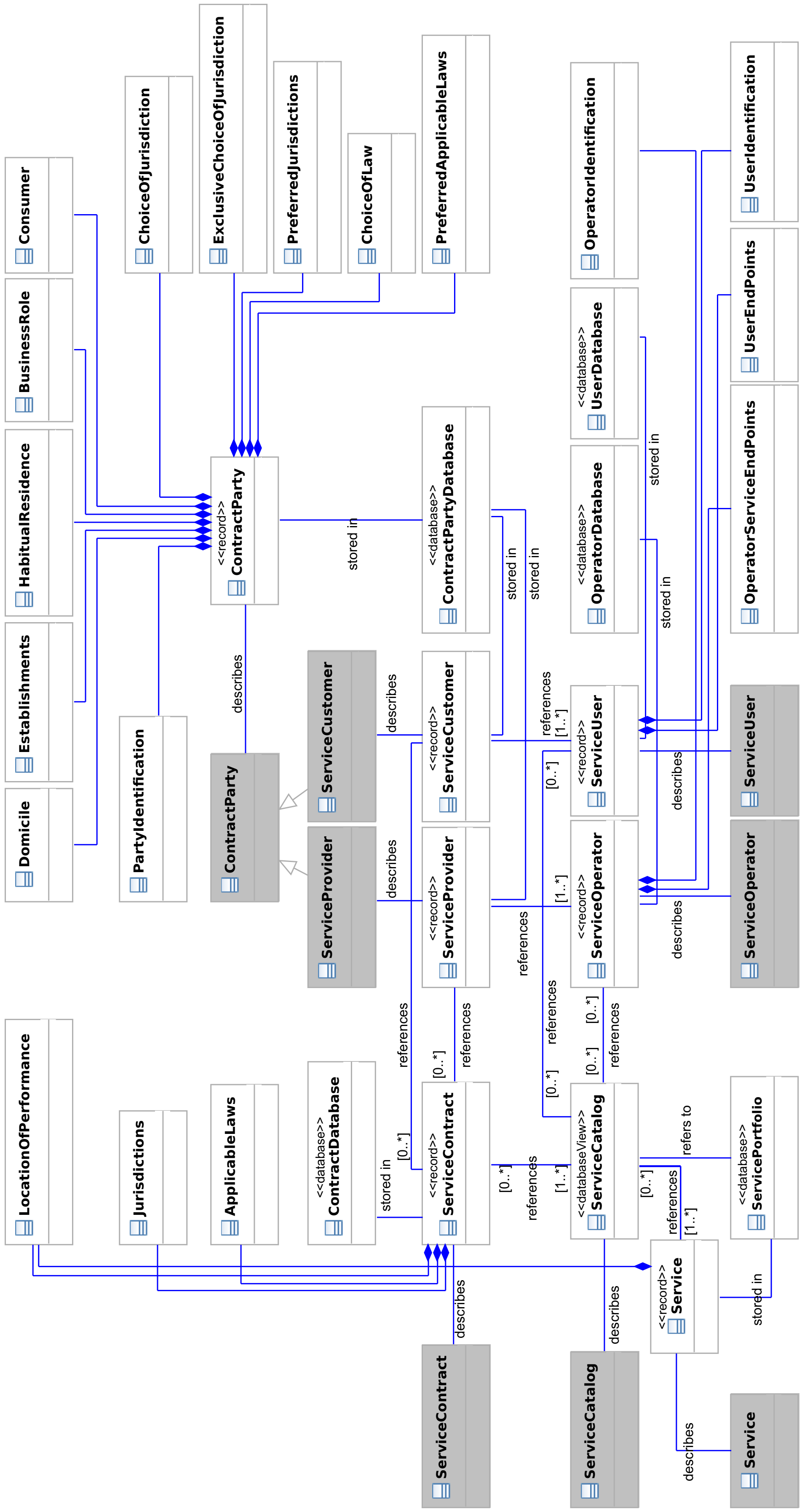


Figure 32: Partial SLAP'N Artefact Model

artefacts that reflect each contract party's characteristics and preferences, while Figure 32 does not elaborate in full detail on artefacts telling about whether both parties actually find agreement according to their mutual preferences so that a contract might see a choice of jurisdiction/applicable law (and if yes, which choice). Similarly, the partial model in Figure 32 includes a generic, *i.e.*, not further differentiated, artefact for jurisdiction(s) and applicable law(s). This is, contrary to those artefacts discussed previously, in line with the two mentioned connecting factors attributed with the respective dimension of result obtained. During the implementation work conducted in IELOS, however, it was found that there are different notions of jurisdiction and/or applicable law to be considered. For instance, notions for unwaivable jurisdiction, jurisdiction for service customer claims and jurisdiction for service provider claims, (non-exclusive or exclusive) chosen jurisdiction, jurisdiction for claims in relation to operation of an establishment, and a notion for general non-exclusive and exclusive jurisdiction need to be clearly separated and handled accordingly. As these aspects are subject to ongoing research, Figure 32 includes generic artefacts for jurisdiction and applicable law at this point.

Finally, those dispute-driven connecting factors of Table 9 do not find representation in the partial information model at all. PIL sources, such as the Brussels I and Rome I regulations, typically contain provisions that presume there is a dispute out of an earlier concluded contract. For PIL, this assumption is a valid one—without a dispute, a court would not have to decide whether it has jurisdiction to hear a case (and if yes, under which nation's material law). For SLAP'N, however, which encompasses an information model reflecting artefacts needed to determine jurisdiction/applicable law at the time of contract formation, no information about any potential future conflict can be taken. The implementation in IELOS ignores this fact, since in IELOS a focus was put on making an implementation available and demonstrating its functionality. This issue is addressed in ongoing research by re-modeling the already modeled PILs in a way that circumvents provisions in dependence of knowledge about a dispute. Accordingly, the inclusion of specific artefacts replacing defendant/claimant and dispute matter is foreseen for future work.

5.4 Preliminary Achievements

SLAP'N has preliminarily achieved its assigned objectives within the determined scope. In particular, a formal modeling of a bilateral service provider/service customer relation for an electronic value-added service has been addressed. This was made possible by the development and documentation of an information model which covers the complete set of required concepts and information artefacts to determine jurisdiction and applicable law for the provider/customer relation envisaged within the frame of an international contract to be concluded. The SLAP'N information model is the outcome of combining two so far separated research works. SLAP'N integrates results from two EMANICS dissertations, one of which is already concluded [24].

Both, concepts and artefacts, are modeled as UML class diagrams. By help of an ongoing collaborative effort with IELOS, it was found that both projects profit from an (already started) extension of their respective informational or functional views on the determination of jurisdiction and applicable law. Consequently, both projects are currently working on a scientific paper to be submitted to the 18th Biennial International Telecommunications Society Conference (ITS2010, conference website: http://www.its2010tokyo.com/index.html).

6 Summary and Preliminary Conclusions

The global outcome of WP8 in EMANICS consists of the development of joint models, economics in network and service management, and insights gained in selected important economic management mechanisms. The deliverable at hand documents key contributions to these areas raised in terms of further developed separate, but overlapping dimensions within the scope of the integrated model for economic management. Figure 33 visualizes

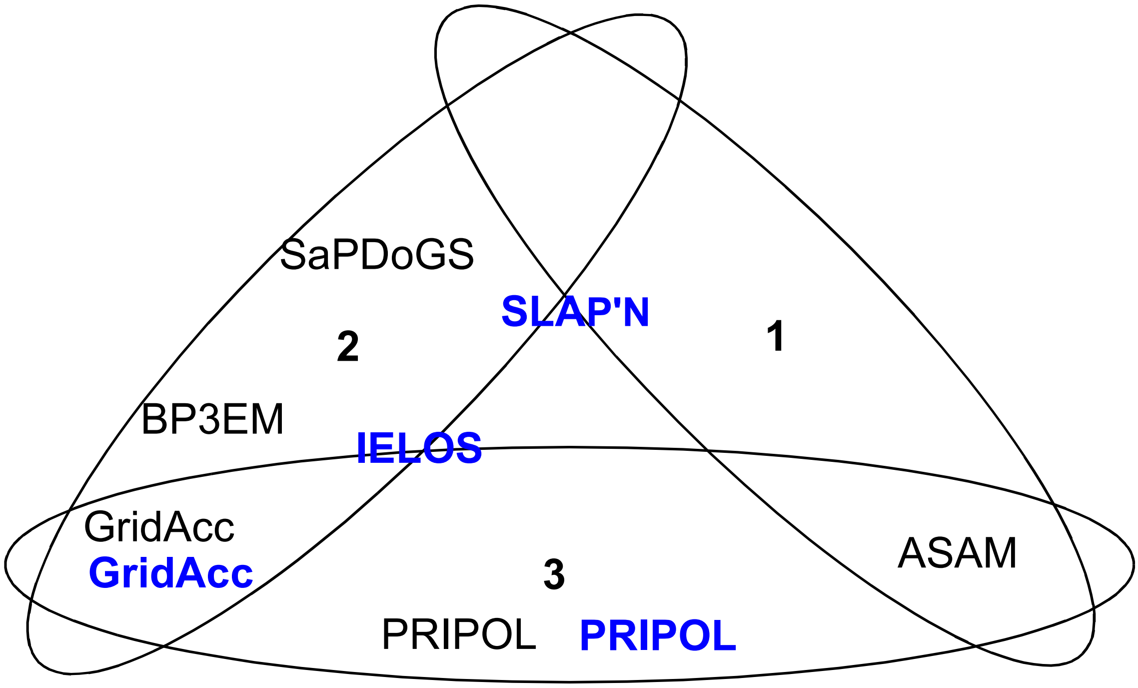


Figure 33: Positioning of WP8 projects According to Economic Management Dimensions

this by positioning previous and current (marked in blue) EMANICS WP8 projects as presented in this deliverable according to key economic management dimensions as follows:

1. Multi-domain Aspects
2. Optimization of IT Service Management
3. Mechanisms

GridAcc focused in 2009 on cost impacts through inter-job effects and resource scarcity in large computing infrastructures such as Grid systems. The project has successfully documented its requirements and scenarios, and it has completed the system design. Accordingly, the relevant set of next steps—some of which have already been started—have been determined to consist in implementing a prototype system and to publish scientific results in the research community. The GridAcc prototype aims to help Grid operators take decisions about the cost efficiency of a considered scheduling algorithm.

The approach adopted by PRIPOL consists in identifying a set of mapping functions between business indicators and parameters of configuration policies. In that way, a system administrator giving specific importance weight to the available business indicators is directly affecting the policies to be enforced to achieve such indicators. PRIPOL has successfully shown by a number of simulations reflecting varying assumptions on user mobility that a business-driven management as foreseen by PRIPOL leads to attractive

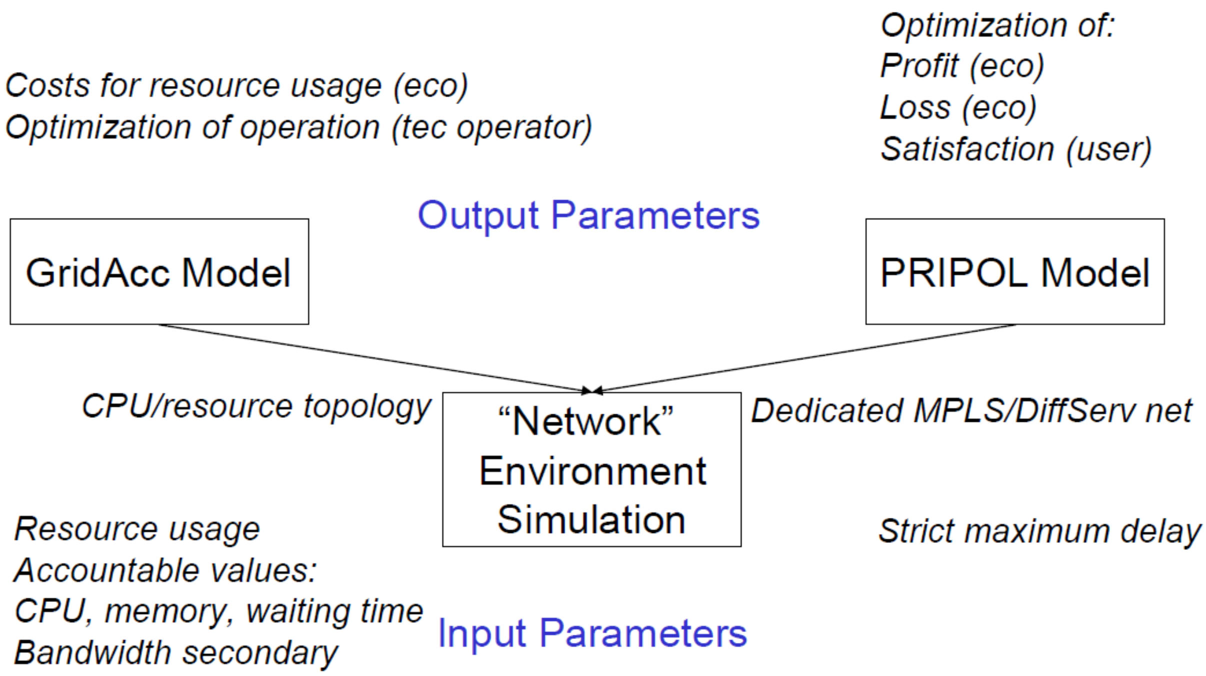


Figure 34: Visualization of Model Inter-relationship between GridAcc and PRIPOL

results for both, users (measured by an increased number of connected users per acces point) and operators (measured by an increased profit).

As seen in Figure 33, GridAcc and PRIPOL, constitute two projects within WP8 which focus on different economic management dimensions in the first place. Nonetheless, Figure 34 provides a valid example to visualize model inter-relationships between GridAcc and PRIPOL. Both projects work on a specific set of input parameters. For GridAcc, example input parameters considered are parameters in relation to Grid resource usage, such as CPU seconds, memory, and bandwidth used. For PRIPOL, a considered input parameter is a maximum value for delay within an MPLS/DiffServ network. Similarly, the respective sets of output parameters in GridAcc and PRIPOL differ from each other. Both sets, however, share the economic management dimension. This may lead to a fruitful adoption of those economic optimizations obtained through business-driven management when porting them to a network simulation environment that considers a topology of CPUs and other Grid resources instead of an MPLS/DiffServ network.

Similar potential for model inter-relationships between all four WP8 projects of 2009 exist. More obvious examples are found betwen PRIPOL and IELOS as well as between IELOS and SLAP'N. A collaborative effort between PRIPOL and IELOS has already led to the submission and acceptance of a joint paper [15] for 12th IEEE/IFIP Network Operations and Management Symposium (NOMS 2010). The respective contribution from IELOS describes the elements required to bridge the gap between business and configuration management in the DiffServ QoS management domain. This has been achieved by elaborating on a bidirectional approach that eases the analysis of business-driven DiffServ management strategies. This approach relies on business indicators that are used by ISPs to derive a network configuration that is in line with high-level business objectives. The set of considered business indicators includes, in particular, *service satisfaction*, *profit*, and *loss* and their influence on both static and dynamic admission control of services. The approach was evaluated through simulation experiments.

Equal potential for model inter-relationship has been identified between IELOS and SLAP'N. Consequently, both projects are currently working on a joint paper. SLAP'N targets a formal modeling of a bilateral service provider/service customer relation for an electronic value-added service. In particular, the development and documentation of an information model which covers the complete set of required concepts and information artefacts to determine jurisdiction and applicable law for the provider/customer relation envisaged is addressed. This bases on the fundamental assumption that service provider and service customer are not located within the same legal domain and/or that a service needs to be provided between different legal domains, so that an international contract is to be concluded. IELOS adopts the same assumptions and addresses a prototype implementation of a rule-based system to determine applicable law according to the EU Rome I regulation. Ongoing work in IELOS is concerned with developing and adapting a refined, more structured implementation technique. This new technique results in higher modularization and improved functionality assessment according to defined use cases. Moreover, it incorporates naming schemes which are consistent with the information model determined in SLAP'N. Finally, the so far considered case of applicable law is currently extended by the respective case of jurisdiction determination.

Those provided cases of a potential inter-relationship between GridAcc and PRIPOL and the ongoing collaborative efforts between IELOS and SLAP'N as well as between PRIPOL and IELOS are examples of inter-relationships of apparently separate projects, each of which focusing on improving selected economic management mechanisms. The dimension of economic management, however, constitutes the essential element that facilitates a successful mutual interchange between these projects. Hence, it is the common economic management model developed in the scope of EMANICS WP8 that provides for a shared solid basis for important extension beyond current state on a separate as well as on a joint basis.

7 Glossary

This section outlines again the major terms, which form the basis of those key multi-provider, and SLM models as well as of those service provisioning concepts described in this document. Additionally, it has been extended by those terms of relevance from D8.2.

- **Agreement**
  A common understanding about knowledge that is shared between two parties. Agreements are often assumed to be about policies or actions (agreements to act) and are often formalized using contracts in which case both parties agree to the terms of a common contract.
- **Architecture**
  An Architecture describes interactions of components of a complex system. Often, Architectures provide a layered or comparably structured view on the respective system.
- **Contract**
  a bilateral bundle of promises between two agents, that is intended to serve as the body of an agreement.
- **Framework**
  A Framework represents a reusable design for a system by describing concepts and structures that give guidance for the execution of a system's complex tasks without

providing strict and mandatory implementation requirements or specifications. A Framework is often less concrete than a Model and described in a natural language rather than by using formal modeling techniques.

- **Model**
  A model is a representation or description designed to illustrate the structure or method of operation of an object, system or concept. In this capability, models are often used to simplify, down-scale and/or abstract from real-world entities.
- **Multi-domain**
  Adjective designating the characteristic (*e.g.*, of a management framework), that more than one administrative domain is involved. These domains often have to establish cooperation agreements on a peer to peer basis, coordinating aspects like configuration management or interaction strategies.
- **Multi-provider**
  Adjective designating the characteristic (*e.g.,* of a management framework), that more than one provider is involved. Multi-provider scenarios usually entail technical, operational and economic or legal cooperation issues between the participants to be solved by agreements.
- **Policies**
  A policy defines a course or method of action selected among alternatives and in light of given conditions to guide and determine present and future decisions.
- **Promise Agreement**
  A promise agreement is a pair of promises between two parties to acknowledge the content of an contract body.
- **Service**
  A service is the entity or unit of work offered by a service provider on behalf of a service consumer who can use the service. In general, a service includes several types of resources, including hardware- and software resources such as computing power, network links, storage capacity, and content, and it may even be composed of several sub-services.
- **Service Catalog**
  A Service Catalogue contains definitions of standard services as well as documentations of customer-specific services. It can be used as a foundation for automated service subscription or for the negotiation of SLAs.
- **Service Level Agreement (SLA)**
  A Service Level Agreement (SLA) defines the terms under which a service is offered to a service customer at a specific Service Access Point (SAP). The SLA includes a set of parameters which specify the service and the QoS under which it is provided (*e.g.*, the amount of bandwidth allocated, the involved session partners, metrics and algorithms that are used to compute SLA parameters), accountable units and the tariff which is used to charge for the service usage. Besides, several other aspects such as penalties or actions, respectively, to be taken if SLA objectives (*i.e.,* guarantees) are violated, trust relationships are part of a SLA.
- **Service Provisioning**
  IT services can be associated with a service life cycle that subsumes the steps from planning to termination of a particular service. A popular simple service life cycle is called Plan-Build-Run, typically rerun after a Change or Improvement step. Service Provisioning mainly deals with the plan, build and change parts, providing the necessary input for run time operations. It is therefore a major part of the service life cycle. More

precisely, service provisioning includes tasks such as planning new services, building the basic infrastructure, SLA negotiation and order processing, identifying adequate resources for service delivery or adapting existing services to specific customer needs, specifying steps for service implementation and service operation, up to dynamic, near real-time service composition out of service modules based on customer requirements.

- **Virtual Organization (VO)**
  A Virtual Organization (VO) is a form of organization abstraction. It is understood as a temporary or permanent coalition of geographically dispersed individuals, groups, organizational units or entire organizations that share resources, capabilities and information to achieve common objectives. VOs can provide services and thus, can take the role of a service provider.

8 References

[1] I. Aib, R. Boutaba: *On Leveraging Policy-Based Management for Maximizing Business Profit*; IEEE Transactions on Network and Service Management, Vol. 4(3), pp. 25-39, 2007.

[2] J. Baliosian, J. Serrat: *Finite State Transducers for Policy Evaluation and Conflict Resolution*; 5th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY 2004), pp. 250-259, June 2004.

[3] E. Borcocci et al: *Admission control algorithm for aggregated pipes service invocation in multi-domain IP environment*; International Workshop on Next Generation Networking, Coimbra, Portugal, 2006.

[4] M. J. Buco et al: *Utility Computing SLA management Based Upon Business Objectives*; IBM Systems Journal, Vol. 43(1), pp. 159-178, 2004.

[5] A.Caporuscio, A. Carzaniga, A. L. Wolf: *Design and Evaluation of a Support Service for Mobile, Wireless Publish/Subscribe Applications*; IEEE Transactions on Software Engineering, Vol. 29(12), pp. 1059-1071, 2003.

[6] X.Chang, D.W. Petr: *A Survey of Pricing for Integrated Service Networks*; Elsevier Computer Communications, Vol. 24(18), pp. 1808-1818, December 2001.

[7] M. Charalambides et al: *Policy Conflict Analysis for DiffServ Quality of Service Management*; IEEE Transactions on Network and Service Management, Vol. 6(1), pp. 1-16, March 2009.

[8] The Council of the European Union: *Council Regulation (EC) No 44/2001 of 22 December 2000 on Jurisdiction and the Recognition and Enforcement of Judgments in Civil and Commercial Matters*; Official Journal of the European Communities, L12, pp. 1–23, January 2001.

[9] ETSI, Universal Mobile Telecommunication System (UMTS): *Selection Procedures for the Choice of Radio Transmission Technologies of the UMTS*; UMTS 30.03 Version 3.2.0, April 1998, (Available:) http://www.3gpp.org/ftp/Specs/html-info/3003U.htm.

[10] The European Parliament and the Council of the European Union: *Regulation (EC) No 593/2008 of the European Parliament and of the Council of 17 June 2008 on the Law Applicable to Contractual Obligations (Rome I)*; Official Journal of the European Union, L177, pp. 6–16, July 2008.

[11] M. Göhner, M. Waldburger, F. Gubler, G. Dreo Rodosek, B. Stiller: *An Accounting Model for Dynamic Virtual Organizations*; 7th IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2007), Rio de Janeiro, Brazil, pp. 241-248, May 2007.

[12] X. Hong, M. Gerla, G. Pei, C.-C. Chiang: *A Group Mobility Model for Ad Hoc Wireless Networks*; 2nd ACM International Workshop on Modeling, Analysis, and Simulation of Wireless and Mobile System (MSWiM), pp. 53-60, 1999.

[13] M. P. Howarth et al: *Provisioning for Interdomain Quality of Service: the MESCAL Approach*; IEEE Communications Magazine, Vol. 43(6), pp. 129-137, June 2005.

[14] D. Karger, C. Stein, J. Wein: *Scheduling Algorithms*; Book Chapter, CRC Handbook of Computer Science, pp. 1-52, 1997, (Available:) http://people.csail.mit.edu/karger/Papers/scheduling.pdf.

[15] J. R. Loyola, M. Charalambides, I. Aib, J. Serrat, G. Pavlou, R. Boutaba: *Business-driven Management of Differentiated Services*; (accepted for publication at) 12th IFIP/IEEE Network Operations and Management Symposium, Osaka, Japan, April 2010.

[16] J. R. Loyola, J. Serrat, M. Charalambides, P. Flegkas, G.Pavlou: *A Methodological Approach towards the Refinement Problem in Policy-based Management Systems*; IEEE Communications Magazine, Vol. 44(10), pp. 60-68, October 2006.

[17] LRZ (Leibniz Supercomputing Center): *Batch & Interactive Jobs and Policies*; October 2009, (Available:) http://www.lrz-muenchen.de/services/compute/hlrb/batch/.

[18] LRZ (Leibniz Supercomputing Center): *Hardware Description of HLRB II*; March 2007, (Available:) http://www.lrz-muenchen.de/services/compute/hlrb/hardware/.

[19] LRZ (Leibniz Supercomputing Center): Statistiken Höchstleistungsrechner (HLRB); February 2009, (Available:) http://www.lrz-muenchen.de/services/compute/statistik-hlrb/.

[20] B. Moore, E. Ellesson, J. Strassner, A. Westerinen: *Policy Core Information Model – Version 1 Specification*; RFC 3060 (Proposed Standard), Feb. 2001, updated by RFC 3460, (Available:) http://www.ietf.org/rfc/rfc3060.txt.

[21] E. Mykoniati et al: *Admission Control for Providing QoS in IP DiffServ Networks: the TEQUILA Approach*; IEEE Communications Magazine, Vol. 41(1), pp. 38-44, 2003.

[22] D. Niyato, E.Hossain: *Competitive Pricing in Heterogeneous Wireless Access Networks: Issues and Approaches*; IEEE Network, Vol. 22(6), pp. 4-11; November-December 2008.

[23] C. Rudd, V. Lloyd (Eds.): *Service Design*; IT Infrastructure Library (ITIL), The Stationary Office (TSO), Office of Government Commerce (OGC), 2007.

[24] T. Schaaf: *IT-gestütztes Service-Level-Management: Anforderungen und Spezifikation einer Managementarchitektur*; (in German), Dissertation, LMU München, Fakultät für Mathematik, Informatik und Statistik, pp. 1-440, November 2008.

[25] B. Stiller, D. Hausheer (Eds.): *Development of a Multi-provider Model, an SLM Model, and Service Provisioning Concepts*; Deliverable D8.2 of EMANICS, pp. 1-93, July 2007.

[26] B. Stiller, D. Hausheer, G. Schaffrath (Eds.): *Definition of Service Provisioning Goals,Economic Impacts, and SLA Management Tasks*; Deliverable D8.1 of EMANICS, pp. 1-114, June 2006.

[27] P. Trimintzios et al: *Service-driven Traffic Engineering for Intra-domain Quality of Service Management*; IEEE Network Magazine, Vol. 17(3), pp. 29-36, 2003.

[28] M. Waldburger, M. Göhner, H. Reiser, G. Dreo Rodosek, B. Stiller: *Evaluation of an Accounting Model for Dynamic Virtual Organizations*; Journal of Grid Computing, Springer, Vol. Online First, No. DOI: 10.1007/s10723-008-9109-9, Netherlands, pp. 1-19, September 2008.

[29] M. Waldburger, B. Stiller: *Automated Contract Formation for Electronic Value-added Services in the Internet—The Case of Bandwidth-on-Demand Contracts in Europe*; 37th Research Conference on Communication, Information and Internet Policy (TPRC 2009), pp. 1-30, Arlington (VA), USA, September 2009.

[30] M. Waldburger, B. Stiller (Eds.): *Definition of a Draft Extended IP Network Management Model*; Deliverable D8.3 of EMANICS, pp. 1-68, January 2008.

[31] M. Waldburger, B. Stiller (Eds.): *Documentation of Final Model and Evaluation of Economic Management Principles*; Deliverable D8.4 of EMANICS, pp. 1-95, December 2008.

9 Acknowledgements

This deliverable was made possible due to the large help of the WP8 team of the EMANICS team within the NoE, which includes this deliverable's authors as indicated in the document control. Many thanks to all of them.

10 Annex 1: Report of IETF 75 (75th Internet Engineering Task Force)

- Date: 27-31 July 2009
- Location: Stockholm, Sweden
- Participants: 1233 registrations
- Participant from UniZH: Fabio Hecht
- Meeting material: slides etc. are available at https://datatracker.ietf.org/meeting/75/materials.html

10.1 Peer-to-peer Research Group (p2prg) Session

10.1.1 P2P mythbustering (draft-marocco-p2prg-mythbustering-01, Enrico Marocco)

Marocco presented the draft, which tries to sort out myths and facts about traffic localization. It is very interesting for SmoothIT. One of the myths is "increased application performance", results from several studies show that application performance increases, but degree varies. Zoran asked why are results so different, Marocco replied it is because scenarios are different, and results seem to vary a lot depending on the scenario, we are here to try to understand why. Zoran: we should then make clear what are the differences. Marocco: yes. Björn Pahlsson asks what is the distribution of downloaded files? Marocco: I don't know, normally it is stated on the paper. We don't have time to go through each of them. Each paper does it in its own way. Marocco presents scenarios in which an increase in local uploads may happen.

He also shows what is the mpact on peering - smaller IPSs like to peer with large ones because they save money. ALTO could be used by them to increase traffic that is sent to them, almost a form of traffic injection.

The transit provider may also redirect traffic to some provider it provides transit to in order to make more money. So, traffic localization is not always win-win. Someone in the audience says he sees those problems as attacks, and thinks that there must be a way to

prove that the underlay topology is being correctly informed. Marocco invited him to bring this to the ALTO meeting this afternoon, since they will be discussing requirements.

10.1.2 Survey on ALTO (draft-rimac-p2prg-alto-survey-00, Marco Tomsu)

Marco presented a taxonomy of different traffic localization proposals, including Vivaldi, Ono, Meridian, iPlane, P4P, Oracle, and IDIPS. He explained how each one works, without making any experiments or drawing any conclusions. The next steps are to produce an updated draft, send it to the mailing list, and ask for adoption as a WG draft.

10.1.3 P2P Group Management (draft-kassinen-p2prg-group-management-00, Erkki Harjula)

P2P needs trust among members, since they form communities to share information. The draft introduces a mechanism to manage peer groups. Each group forms its own overlay. He explained the algorithms for group creation and management. There is also a main overlay, containing group meta data. He showed sequence diagram for joining and leaving, group removal, etc. There are many open issues, like bootstrapping. Future work includes secure group management.

A comment from Thomas Schmidt stated that surveys are more adequate as RG items, instead of a particular approach, which is what was presented.

10.1.4 BitTorrent Measurements: Challenges of Locality Promotion (Zoran Despotovic)

Zoran presented the BitTorrent swarm measurements. He showed a table comparing several BT measurement efforts. He focused on as topology, which was not so interesting before as it is now. He showed the distribution of torrents for contents with different languages.

The audience questioned whether he considers ASs according to the IP addresses they are registered. Zoran: yes.

Graphs showed that, according to language, AS distribution is different, Spanish content is very AS-concentrated, while English content is more diverse. He showed another graph where AS hop distance is a better metric than binary locality. There are some ASs that will end up worst off. We should invest time to study those cases. Marocco sees a lot of works in the same direction, would be nice to coordinate and agree on methodology. Zoran: makes sense. Reinaldo Penno: should detail parameters used in simulation.

10.1.5 Introduction of ALTO-like Activity (P2P Net Experiment Council in Japan, Satoshi Kamei)

In Japan, they are shaping traffic based on "Guideline for Packet Shaping" (by ISP, CATV, Telecom associations). P2P traffic is still dominant (60% in 2008). He showed the network cost structure in Japan. The bottleneck is moving from access to the backbone, since access bandwidth is increasing substantially. He showed network neutrality and competition policy in Japan, they have a WG that defines it. They see P2P as a problem but also important solution for scalability. They made an experiment with something like ALTO (Hint Server) and dummy nodes. He compared peers choosing same ISP or same region, and showed in graphs that the distance that traffic has to travel decreases by 20-40% using the Hint Server. They intend to extend the experiment, making it compatible with ALTO requirements. He also gave some suggestions to ALTO: hierarchical structure is useful,

providing statistics is important to see what is the improvement, challenge is cache/NAT solution.

10.1.6 Swarm Analysis: 7 Days of Two Popular Swarms (Vijay Gurbany)

They studied 2 swarms, one in English (movie Push) and the other in Cantonese (movie Sniper). Both swarms had very similar number of peers and numbers of countries. He showed the attenuation rate of each torrent, decay is gradual for sniper, quicker for push. Peer time (duration) seems to be power law distributed. He looked for ASNs, and the number is similar for both movies, at approximately 2830. About 200 ASNs had more than 100 peers. Then he showed peers per country, US leads both, distribution is also power law. He then showed peer/ASN/country. The US then falls several positions, NO, GB, BR, IN lead. The next steps are to continue mining, repeating the experiment with more localized languages (Russian, Hindi etc.). Rich Wound states that Comcast has 22 ASs, while Verizon has only 1, but is 50% the size of Comcast, so he should normalize the results according to ISP size.

I then asked why China is not listed in the top countries for the Cantonese torrent, but he did not know why, hasn't analyzed this.

10.1.7 P2P networking for live streaming (Omer Luzzatti)

They have developed a hybrid P2P/CDN network for video streaming with 3 million downloads, 300K connected clients in average, 70K viewers at peak. Watching habits are just like TV, overall 31 min/day average viewing time, for sports it is 98 min/day. He showed live information about the P2P network, how peers were donating bandwidth. He shows also the live delay, in average only 1 second. In the future, they intend to try and synchronize them better. He shows quality and bitrate differ according to content; news are transmitted at 500-800 kb/s, sports at 800-1500 kb/s. Content owners are very concerned with delay, must be < 10 seconds, and between viewers max 2 seconds. On average, application receives 90% of packets, and application layer uses redundancy (PET and network coding) to reconstruct the stream. They use SCTP, which is a TCP-like congestion control over UDP, with no retransmissions, using Forward TSNs. He shows results of viewing quality, without P2P clients receive in average 86.5% quality, while using 75% P2P boosts reception to 98.8% in average. Using P2P, they could eliminate CDN servers in Australia and Israel, which were very expensive but important to sustain QoS. Average user upload is only 180kb/s, so it is impossible to rely on them. To solve the problem, they take some bandwidth from non-viewers that leave the application running. He also presented some concerns of the content owners. The system must prevent "their" viewers from contributing to other owners, for example. Operators may impose several restrictions and legal rules, e.g. taxes between countries for PPV content. Somebody questioned what kind of topology they use. He said the client requests video to server, and the server replies with a set of peers from the same geographical zone. Peers constantly replace neighbors that are not good, especially in the beginning.

10.2 Application-Layer Traffic Optimization (alto) WG Session

First, the WG status was reviewed. The problem statement draft is in last call. The requirements draf is in progress.

10.2.1 Requirements (draft-ietf-alto-reqs-01, Sebastian Kiesel)

There are some important changes in the new version of the document. The Host Location Attribute has been replaced by three new terms: Host Group Descriptor (HGD), Rating Criterion (RD), and Host Characteristics Attribute (HCA), in order to make things clearer. Sebastian lists HGD requirements, e.g. it must include IPv4 and IPv6 prefixes. It must support ALTO client in peer and tracker. He raises the question of whether the service discovery will redirect the client to the right alto server or the client should contact the server in its own network, which would redirect the query to the right server (iterative or recursive). There was then a discussion, some people were saying iterative is simpler to define and implement, but some think it is difficult for alto servers to know one another; some think that an alto server may know whether it is congested. Reinaldo Penno questions what is the meaning of the "right" ALTO server. Lars Eggert thinks it is going too far as a requirement, it should not define exactly how things must be, only what it should provide. Authentication requirement was slightly changed but only for clarification. Question: have you consider case that client uses an anonymity server? Answer: it is not defined at the moment.

10.2.2 ALTO Protocol Merged Proposal (draft-penno-alto-protocol-03, Reinaldo Penno, Richard Alimi)

The draft is a merged proposal of all other ALTO proposals. Reinaldo presents each of the proposals: P4P, InfoExport, Proxidor, Query/Response, ATTP (client feedback included!). The proposal also refers to the "my-Internet view". Lars Eggert asks whether the my-Internet view is service specific; Reinaldo answers that it could be, peers within the same group must have the same view. Lars also asked whether a peer may only query information about itself or also between 2 other peers - Reinaldo answers that both are possible with his protocol. There was then a discussion about costs and metrics, financial and technical, and whether an ISP would be willing to provide information about other peers. Richard then presents the protocol, it uses HTTP, REST-ful API, and XML encoding. A question from Volker Hilt: there are many ways to do things in this protocol, the implementation of the server and client will be rather complex. Richard says that must be discussed. Next steps are to discuss whether to adopt the document as a WG item, and discuss on possible extensions. A person from Orange thinks that the protocol defines too many ways to do the same thing; also that it protects user privacy but not ISP. Richard answers the the ISP discloses what it wants.

10.2.3 Feedback-based Client Protocol (draft-despotovic-alto-feedback-cp-00, Zoran Despotovic)

Zoran presents the draft, stating that feedback from a client to the server would complement the ALTO protocol. He also shows an example where client feedback would be useful. There were no questions asked.

10.2.4 ALTO Server Discovery (draft-song-alto-server-discovery-01, Marco Tomsu)

Marco merged 2 previous drafts, wang-alto-discovery and song-alto-server-discovery, which were all existing service discovery drafts. It does not present many technical details. There are several mechanism options, as presented in the last meeting, but he also added Geopriv as an extra option. Next step is to adopt draft as WG item. Lars Eggert: how to make it work with DHCP? If client is behind NAT, how to reach provider with broadcast? Jon: at the moment, most would block, but Geopriv is looking into how to solve it.

10.2.5 BGP-based ALTO Service (draft-racz-bgp-based-alto-service-00, Zoran Despotovic)

The question he tries to answer is how to get locality information. The assumption is that the ISP runs the ALTO server. Zoran shows relevant BGP attributes for ranking. Shows then how to use the parameters to rank. David (?) thinks this is an interesting direction. Stefano Previdi finds the approach extremely dangerous, since it uses attributes with different semantics, so he has to be careful not to mess with the routing when changing stuff for alto. Somebody suggests he could also consider IGP.

10.3 Low Extra Delay Background Transport (ledbat) WG Session

10.3.1 Low Extra Delay Background Transport (draft-shalunov-ledbat-congestion, Stanislav Shalunov)

The document has not changed since the last IETF. Stas starts reminding the algorithm proposed already. The receiver side does not change much, but the sender side must be aware of the delay. The type of framing was questioned, whether TCP or UDP. Stas says that his congestion control works with any protocol, but today he has it over UDP. In the future, a modified TCP version would be possible, or SCTP or DCCP CCID or SCTP. Late-comers could have an advantage, since they had not had time to know the real minimum delay. Richard Woundy (Comcast) asks whether there is any study that proves that what he says is generally true, and if technology changes, e.g. spinning disk to SSD, it is still true. There are no studies that prove this yet, and disk seek is one example of what a computer can do, but there are other things that may interfere with the delay. Lars Eggert is worried about cases in which things get worse, but cannot see any. lars: check RFC 5148, it is quite related (measurement of jitter in MANETs). The algorithm has 2 important parameters: gain and target. What should be the target? 25ms? 5 ms? The values in the draft just work, but there is no proof on whether they are optimal. The choice of gain is more arbitrary, it controls only how fast it gets to the target. Gain of 1 was chosen arbitrarily. Human perception is a good basis. Discussion about when would the delay be combined with something else, e.g. an HTML page gets references to load other elements in a potentially cascading effect. Finally, the WG decided to adopt draft as WG item by 18 hands in favor, zero opposing.

10.3.2 LEDBAT Practices and Recommendations (draft-penno-ledbat-app-practices-recommendations, Reinaldo Penno)

Reinaldo, the author, got sick, so a substitute presented. He just read all slides. Question from Stas: how much should this describe and how much should it recommend? Lars Eggert: they wanted to make recommendations. Bob Briscoe: first to educate and explain; to recommend would be difficult. There was then a large discussion on whether multiple TCP connections is beneficial or whether it justs increases overhead and congestion. At the end, people were asked to raise hands to adopt the document as a WG item. Zero were in favor, two were against. The conclusion is that is needs more work but there is nothing fundamentally wrong with the document.

10.3.3 A Survey of Lower-than-Best Effort Transport Protocols (draft-welzl-ledbat-survey, M. Welzl)

The intention is to avoid reinventing the wheel by surveying protocols that could solve the LEDBAT problem. He describes several protocols. Delay-based: TCP Vegas, Reno, Nice,

TCP-LP. Non-delay-based approaches: 4CP, MulTFRC. Application-layer approaches: receive window tuning, BITS by Microsoft. Would it be nice to do it at the application layer? Yes, because it does not need several per-transport protocol specs. This is under consideration for adoption as WG item. 18 hands were in favor, zero opposing.

10.3.4 Home Gate (Iljitsch van Beijnum)

Iljitsch asked for 5 minutes to make a short presentation about buffering in home gateways. A Bar BOF was held Monday on home gateways, called HOMEGATE. They are trying to discover how much buffering should be done in Home Gateways. How much buffering is appropriate for slow links? It is hard to define, with 128 kb/s even 2 packets = 200 ms (too much). He thinks 10 packets is OK for fast links, but too much for slow links. He asks for feedback at www.ietf.org/mailman/listinfo/homegate. Stas disagrees, and thinks a small buffer will make the job harder for LEDBAT; we should make it work for what we have today and particular technologies. Comments: sometimes they buffer to burst when it's their time to use the shared link;LEDBAT traffic should not cause delay, but not fix all existing delay problems.

10.4 DECoupled Application Data Enroute (decade) Bar BoF (unofficial)

DECADE was an informal meeting with objective to discuss the possibility of chartering a new WG. The main idea is to discuss a way to move P2P applications data plane into the network and integrate it with P2P applications seamlessly. It would help to save the last-mile bandwidth problem caused by P2P applications. They also intend to discuss to a possible protocol to access in-network storage. They also intend to support client/server technology, not only P2P. There are no drafts yet, so most of the time was allocated for discussing. Question: caches exist and there is CDN logic, why not just use whatever they do? Answer: that's one possibility. They must raise requirements and check against CDN characteristics.

10.4.1 Problem statement (Haibin Song)

P2P traffic is significant, and it puts pressure on network, especially on access link (last mile). Asymmetric uplink puts pressure on uplink. Discussion about this, Rich Woundy agrees this is the case for Comcast, but not necessarily for others. A comment say that he has to make clear what kind of application he has in mind. P2P can be many things.

The problems of existing solutions are: tight cache/application protocol coupling, tight signaling/data plane protocol coupling in cache. P2P application can try to use the cache and, if not possible, fall back to its own protocol. Content discovery is crucial. Question: what is the difference between your protocol and RELOAD? Answer: RELOAD is for resource discovery.

The proposed scope is to define a standard, lightweight control between in-network storage and P2P application. Out of scope would be integration with existing P2P applications. He presents also initial requirements.

10.4.2 Survey on Network Storage (Ning Zong, Richard Alimi)

Rich presents system components: discovery, access protocol, operations, access & resource control etc. He presents pros and cons of NFS, web cache (HTPT uses HTTP caches to cache P2P files/streams). Transparent P2P cache, non-transparent P2P caches,

CDNs, Windows Azure, Oceanstore, ISCSI. Access control with many P2P peers is difficult and a common problem.

10.4.3 Sailor (Richard Alimi)

Rich presents a possible solution to decade by using data lockers. He presents the data lockers idea, which Richard Yang has already presented in San Francisco. He presents scenarios, (1) data lockers at p2p peers, (2) lockers at the content publishers. Presents architecture, it seems to be server/client. Users have control about allocation of their data lockers. He also presents the locker access protocol (LAP), requirements, authorization. Finally, he presents simulation results, a P2P streaming application, with Sailor the delay is shorter.

10.4.4 BranchCache (Yu-Shun Wang)

Yu-Shun presents a new feature of Windows 7. The problem is to optimize WAN links to branch offices, since data tends to be centralized. The solution is to try to find the data at the branch first. The deployment is easy, instrument HTTP(S), SMB, Bits, transparently to the application. The client uses an HTTP header to indicate that it is BranchCache-aware. They use a segment/block notion similar to LiveShift. The server hashes blocks and sends the hash list to the clients, which see if they need to get the any parts. The security is achieved with a key infrastructure.

Rough consensus was reached to decide whether DECADE has a problem that the IETF could tackle. But it will be good to understand what specific problems that today are not solved.

10.5 Technical plenary

Spotify (Gunnar Kreitz)

Spotify is a lightweight on-demand streaming system, with a large catalog of licensed music. The main idea is to make it more convenient than pirate sources. A free version is ad-funded, users can pay to get rid of ads. Protocol is proprietary, bit rate is 160-320kb/s. The stream comes from peers, servers, or CDN. The network protocol is designed for random-access streaming. They use mostly TCP (!) and everything is encrypted. They multiplex everything on a single TCP connection between hosts, allowing more control. Latency is an important metric to be minimized to provide good QoE. To play a track, the server must be contacted first, then it keeps on switching from getting content from server and peers (when possible). A mesh structure is used in P2P, nodes have fixed maximum degree, with neighbor eviction by heuristic evaluation of utility. He thinks IETF is tackling important P2P challenges, such as developing protocols for streaming itself and ALTO-like. The big challenge is to tackle mobility. Locality awareness is also important.

Question: what is the importance on using P2P? Is bandwidth cost important compared to license? Answer: It does help a lot, since it is ad-funded and costs would be too high to maintain if not P2P. Question: ipv6 support coming soon? Answer: IPv6 would solve many problems but it is not deployed enough.

10.5.1 Peerialism (Andreas Dahlstrom)

Peerialism offers P2P for large-scale distribution and storage. P2P 1.0, as he calls it, is based on random, unstructured traffic. P2P 2.0 takes care about the whole business ecosystem, it must be topology-aware, have an efficient peer resource management, doing

local rather than global optimizations. A central server takes care of network optimization. Second example of P2P 2.0 is the new Pirate Bay. They are developing Open Tracker 2.0, which is locality-aware. Idea is to see 20-50% less traffic loads, and 30-150% higher download speeds. It requires no chances to BT clients. Open Tracker 2.0 is to be launched is September 2009. Locality information comes from BGP data and also from peers, to their locality algorithm, which computes a global network map, with links and their quality. Locality in BT is only good until a certain spot, which was found to be around 90%.

Question: is there any relationship between them and P4P? Answer: he likes P4P but the problem is that operators have to be part of it. Perialism works even without P4P. Comment: it is difficult to get a map of the Internet, even with many points. Question/comment: P4P is not an academic project, operators will participate. Answer: he just does not want to rely on that, it may take too long time. Question: he talked to operators about BGP, and problem is that some have 1 ASN and others have many ASNs, do you take that into account? Answer: yes, they have possibility to put costs at a router level. Question: it seems that there is interest from ISPs to help, e.g. by deploying caches. Answer: they have not looked into caches. It seems that ISPs want caches to be HTTP-compatible.

10.5.2 Network Neutrality (Barbara van Schewick)

Barbara presents some slides about network neutrality. What are the incentives to block some content or service? Not usually, but there are cases of ISPs blocking e.g. VoIP, since it threatens a revenue source. Or making people stay longer in their website to see more ads. Comcast offers basic access without VPN for a cheaper price, since people who use VPN are business and therefore willing to pay more. She gives example that supermarkets may not have your favorite kind of chocolate, but you can go to the next store. It is not so easy to switch providers, if at all possible, due to lack of options. If not blocked, packets can also be discriminated with regards to QoS.

10.5.3 Network Neutrality and the IETF (Mark Handley)

Why should the IETF care about network neutrality? Why should it not? It is economic and legal, it must work in different countries. The problem is blocking, rate-limiting or prioritizing packets based on service or destination. DPI is not really a choice, due to privacy issues and the fact that it does not really work with tunneled or encrypted traffic. He likes IETF efforts like LEDBAT, Multipath TCP, multi-server HTTP, and diffserv. There was then a large open-mic discussion on network neutrality and the role of the IETF.

10.6 Peer-to-peer Streaming Protocol (ppsp) Bar BoF (unofficial)

PPSP is trying to charter a WG to define a protocol for P2P Streaming.

10.6.1 Problem Statement (draft-zhang-ppsp-problem-statement-04, Yunfei Zhang)

Yunfei shows that P2P streaming is becoming very popular and standards are needed. He shows use cases for P2P and CDNs. There are several parts where a protocol could be developed. He proposes the scope: to evaluate existing architectures, standardize data format (communication and information layer) and standardize message in communication layer. He shows proposed tasks for the proposed WG.

Question/comment: he could use rendez-vouz finding from P2PSIP. Question/comment: would the P2P companies want to use a unified protocol? Answer: PPLive participated in the mailing list, but they did not say they will implement the protocol. Discussion about what

is necessary to charter, seems that it must be clear that there is interest from industry to collaborate to standard and adopt it afterwards.

10.6.2 Chunk Discovery for P2P Streaming (draft-zong-ppsp-chunk-discovery-00, Ning Zong)

Chunk discovery allows peers to publish streams that they have and lookup streams that other peers have. Ning shows an example sequence diagram of a basic P2P streaming system. Next steps are to evaluate how different applications perform chunk discovery.

Comments say that the problem to be solved is not clear. Some background on how live streaming works would be beneficial.

10.6.3 Protocol Analysis of PPlive, PPStream and UUSee by Interent Measurement (draft-zhang-ppsp-protocol-comparison-measurement-02, Yunfei Zhang)

Cancelled due to lack of time.

Instead, discussion on whether this effort should become a real BoF at the next IETF. There seems to be no strong opposition, though there is lots of work to be done. He will send a formal BoF proposal.

11 Annex 2: Selected Cooperation Work

11.1 Business-driven Management of Differentiated Services

Authors:

Javier Rubio-Loyola, CINVESTAV

Marinos Charalambides, University College London

Issam Aib, University of Waterloo

Joan Serrat, Universitat Politècnica de Catalunya

Raouf Boutaba, University of Waterloo

Abstract:

Several intra- and inter-domain quality of service (QoS) provisioning mechanisms for IP networks have been researched and developed using Differentiated Services (DiffServ) technology. However, the incremental efforts needed to manage DiffServ networks from a business-oriented viewpoint have received relatively little attention. This paper addresses this gap and presents a framework for achieving business-driven QoS provisioning in DiffServ over MPLS networks. We provide a rich model of SLA and business indicators as well as reasonable mapping functions that define, with key degrees of importance, the impact of business indicators over service management policies. Business and service indicators are used to control staticand dynamic admission of services. The paper advances the state of the art by considering the influence of business-level objectives on the policy refinement process. We evaluate and discuss the effectiveness of our approach through a simulation environment that we developed over OPNET.

Table of Contents:

- Introduction
- DiffServ QoS Management

  - → DiffServ QoS Management
  - → Static AC for DiffServ QoS Management
  - → Dynamic AC for DiffServ QoS Management
- Business-driven DiffServ QoS Management Framework
  - → SLA Template
  - → Model of Service, SLA, and Business Indicators
  - → Service Management Objectives
- Analysis of Business-driven Policies
  - → Policy Parameters from BIs in the SLS-S Component
  - → Policy Parameters from BIs in the SLS-I Component
- Experimental Analysis
  - → Services and Network Topology
  - → Business-driven Reaction to Traffic Fluctuations
- Related Work
- Concluding Remarks and Future Work
- Acknowledgment
- References

Publication:

J. R. Loyola, M. Charalambides, I. Aib, J. Serrat, G. Pavlou, R. Boutaba: *Business-driven Management of Differentiated Services*; (accepted for publication at) 12th IFIP/IEEE Network Operations and Management Symposium, Osaka, Japan, April 2010.

Business-driven Management of Differentiated Services

Javier Rubio-Loyola[1], Marinos Charalambides[2], Issam Aib[3], Joan Serrat[4], George Pavlou[2], and Raouf Boutaba[3]
[1]CINVESTAV, [2]University College London , [3]University of Waterloo, [4]Universitat Politècnica de Catalunya
[1]jrubio@tamps.cinvestav.mx, [2]{M.Charalambides, G.Pavlou}@ee.ucl.ac.uk, [3]{iaib, rboutaba}@uwaterloo.ca, [4]serrat@tsc.upc.edu

***Abstract* — Several intra- and inter-domain quality of service (QoS) provisioning mechanisms for IP networks have been researched and developed using Differentiated Services (DiffServ) technology. However, the incremental efforts needed to manage DiffServ networks from a business-oriented viewpoint have received relatively little attention. This paper addresses this gap and presents a framework for achieving business-driven QoS provisioning in DiffServ over MPLS networks. We provide a rich model of SLA and business indicators as well as reasonable mapping functions that define, with key degrees of importance, the impact of business indicators over service management policies. Business and service indicators are used to control static and dynamic admission of services. The paper advances the state of the art by considering the influence of business-level objectives on the policy refinement process. We evaluate and discuss the effectiveness of our approach through a simulation environment that we developed over OPNET.**

Keywords:* *business-driven management, DiffServ, QoS management, policy, Business Indicators

I. INTRODUCTION

Differentiated Services (DiffServ) have been proposed as a scalable approach for providing Quality of Service (QoS) in IP networks. The core philosophy is grouping traffic with similar QoS requirements into a limited number of service classes, allocating bandwidth to these classes, and differentiating their forwarding treatment throughout the network. While different QoS levels can be provided, the absence of advanced control (based on, for example, pricing, service admission, etc) can result into resource starvation and network congestion.

A Service Level Agreement (SLA) is a contract signed between two or more parties relating to a service relationship that sets a clear measurable common understanding of the role each party plays [1]. A party role represents a set of rules defining minimal service level expectations and service level obligations it has with other roles and at which constraints. Constraints normally include contract scope (temporal, geographical, etc.), the agreed upon billing policies, as well as the expected behaviour in case of abnormal service operation [1]. SLA violations, due for example to network congestion incur negative impact on the business value of a service provider and need to be minimized.

Management plane functionality [14] is needed to support end-to-end QoS based on Service Level Specifications (SLSs). Network congestion prevention and resolution, service subscription and invocation control, and dynamic traffic engineering functions have been the centre of study in intra-domain [20] and inter-domain [12] management solutions. Although these solutions demonstrate how services can be delivered with QoS guarantees, the incremental efforts to elevate their business value have remained largely unexplored.

This paper addresses the need to bridge the gap between business value and configuration management. It provides a comprehensive review of the necessary elements to realize business and QoS management for DiffServ networks, and describes an integrated framework that eases the analysis of business-driven DiffServ management strategies. In contrast to prior work that applied business-driven techniques to drive IT and SLA management for data centres [3][5][7], we aim here to tackle the issue of QoS management in DiffServ/MPLS networks. We describe an approach to control relevant business indicators (BIs) and evaluate their relationships with service management objectives and policies. We focus on static and dynamic Admission Control (AC) of service subscriptions and extend previous work on policy refinement [18] by considering the influence of BIs when generating appropriate configuration policies. This is realized by defining a set of mapping functions that take into account the impact of BIs over service management policies. Administrator assigned weights of importance are given to each BI and are then used to derive appropriate policy parameters. We demonstrate, through simulation scenarios, the effectiveness and practicality of this approach, highlighting the issues necessary for its realization.

The paper proceeds as follows. Section II presents background information on DiffServ QoS management. Section III describes our business-driven framework for QoS management in DiffServ/MPLS networks. Section IV analyzes the business-driven policies. Section V provides experimental analysis and Section VI reviews related work. Finally, Section VII concludes with insights on future work.

II. DIFFSERV QOS MANAGEMENT

A. *DiffServ QoS Management*

QoS provisioning in DiffServ networks involves a range of management operations, from traffic engineering and Admission Control (AC), to dynamic management of resources. Several frameworks have been proposed for this purpose that mainly stemmed from European collaborative research projects including TEQUILA [20], MESCAL [12], and ENTHRONE [6]. All frameworks propose the use of a general model depicted in Fig. 1, where the QoS management goals are realized by three distinct functional blocks: *Service*

Management, *Traffic Engineering*, and *Policy Management*. The first is responsible for agreeing the customers' or peer domain's QoS requirements in terms of *Service Level Specifications* (SLSs). Traffic Engineering fulfills contracted SLSs by deriving network configuration. Policy Management guides the behavior of the two aforementioned blocks with a set of policies to reflect the high-level goals and objectives.

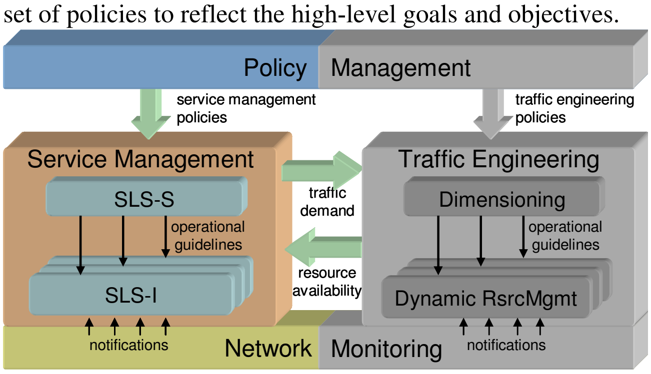


Fig. 1. Policy-driven QoS Management Framework. The business-driven approach focuses on the left part of the figure.

The business-driven approach presented in this paper focuses on the functionality provided by the Service Management block. This is realized by the *SLS-Subscription (SLS-S)* and *SLS-Invocation (SLS-I)* components. The former has a centralized off-line functionality and performs admission control on subscription requests based on resource availability provided by the Traffic Engineering system, whereas SLS-I is distributed across ingress routers and performs dynamic invocation of already subscribed SLSs based on the network runtime state, following operational guidelines provided by SLS-S. Both components assume an MPLS-enabled network.

Another issue relevant to the business-driven approach is the identification of incidents affecting the business value. Incidents like "max contracted service rate crossed" or "network link down" highlight situations that would impact the business, and for which prompt actions may be necessary.

B. *Static AC for DiffServ QoS Management*

The policy-based subscription logic performs static admission control of the number and type of service subscriptions in order to avoid network overloading while maximizing subscribed traffic. The policies to achieve this objective [9] are shown in Table I.

Actions P1.1 and P1.2, use the notion of service satisfaction and quality levels [13] to set the relevant parameters per *QoS Class* (QC). Satisfaction parameters define factors $\in [0,0.5]$ to derive the rates at which a service is considered *almost* (AS) or *fully* satisfied (FS). They operate on an initial average rate (SR_{AVG}) per service type that is suggested by the provider. An increasing AS factor produces lower rates ($SR_{AVG}*(1\text{-}Fctr_{AS})$). An increasing FS factor results to higher rates ($SR_{AVG}*(1+Fctr_{AS})$). The *Overall Quality Level* parameter (OQL $\in$ [-1,1]) is analogous to the confidence level with which an SLS enjoys the agreed QoS. High priority QCs have an OQL close to 1.

TABLE I
SLS-S POLICY ACTIONS

ID	Policy action	Description
P1.1	setAlmstSatisf(QC, $Fctr_{AS}$) setFullSatisf(QC, $Fctr_{FS}$)	Sets the almost and full satisfaction factors per QC
P1.2	setQltLvl(QC, OQL)	Sets the overall quality level per QC
P1.3	setUpprLmt(TT, SU)	Sets the RAB upper limit per TT
P1.4	setACStrg(QC, TD)	Sets the AC strategy for accept/reject decision per QC

Resource availability is tracked using the *Resource Availability Buffer* (RAB), which maintains the aggregate demand of subscribed SLSs per *Traffic Trunk* (TT). As illustrated in Fig. 2, the buffer up to R^{w}_{min} can be used with high confidence even at times of congestion, whereas the area between R^{w}_{min} and R_{max} is risky because the network cannot provide guarantees [13]. Section V-A shows how the values of R^{a}_{min}, R^{w}_{min} and R_{max} are derived. Policy action P1.3 (Table I) sets the upper limit – *Subscription Upper* (SU) – as a percentage of the buffer between R^{w}_{min} and R_{max} for accepting new subscriptions and defines the level of associated risk in satisfying the QoS requirements.

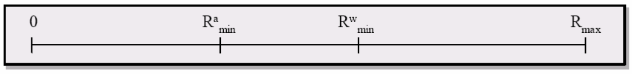


Fig. 2. The resource availability buffer (RAB)

Upon a new subscription request, the overall *Traffic Demand* (TD) for a TT is updated by policy action P1.4 (Table I) taking into account the rate of the candidate service. The subscription is accepted only if TD $\leq$ SU in the RAB. Based on the satisfaction factors set by policy P1.1, TD can range from TD_{min} or TD_{max}, which are derived from AS and FS factors respectively. For the same SU, the use of TD_{min} results in more accepted subscriptions than with TD_{max}, due to the lower service rate.

C. *Dynamic AC for DiffServ QoS Management*

In contrast to the static nature of the subscription component, the service invocation logic is based on run-time events to regulate traffic entering the network. The policies used here are shown in Table II and are enforced on SLS-I. They perform dynamic admission control on the number of active services, as well as on the volume of admitted traffic.

Policy action P2.1 defines a threshold that signals *Target Critical Levels* (TCL) of traffic and $\in [R^{a}_{min}, R_{max}]$. The closer it is to R^{a}_{min} the earlier a notification is issued, whereas values close to R_{max} result in delayed proactive actions.

The run-time operation of SLS-I is triggered by TCL crossing alarms, which activate policy actions P2.2 and P2.3 for adjusting the *Service Rate* (SR) and the dynamic *Admission Control Threshold* (AC_{th}) of a TT. $AC_{th} \in [R^{a}_{min}, R_{max}]$ controls invocations of already subscribed services. The lower it is, the less the chances are of an incoming SLS being successfully invoked. A new service request will be accepted only if the current utilization of the relevant TT together with

the average rate of that service does not exceed AC_{th}. SR ∈ [AS, FS] is the level of satisfaction allocated to a TT. The policy below encodes action P2.2 into policy specification following the Ponder format [10] and sets SR to the almost satisfied level upon an upward TCL threshold crossing event.

```
inst oblig /policies/sls-i/P2.2 {
        on     TCLAlarmRaised(up, TT);
        subj   s = sls-iPMA;
        targ   t = sls-i/servAdjustMO;
        do     t.setSR(tt1, srAS);
        when   duration(08:00-18:00);
}
```

TABLE II
SLS-I POLICY ACTIONS

ID	Policy action	Description
P2.1	setTCL(TT, TCL)	Sets the target critical level threshold wrt RAB per TT
P2.2	setSR(TT, SR)	Sets the service rate of a TT
P2.3	setACth(TT, AC)	Sets the admission control limit wrt RAB per TT

III. BUSINESS-DRIVEN DIFFSERV QOS MANAGEMENT FRAMEWORK

This section provides a business-driven QoS management framework for DiffServ/MPLS network providers. First we consider the types of SLAs to which this model is restricted.

A. SLA Template

We define an SLA S_i for customer *i* as a package of network services $s_i^1, \ldots, s_i^k$ offered as a bundle during a specified period of time with specific quality parameters.

S_i = *{type, schedule, service package =* { $s_i^1, \ldots, s_i^k$ }, *availability, quality, quality degradation refund policies.}*

A network service (FTP, VoIP, Web, etc.) has the form:

{service type, unitary cost, throughput [average, min, max, window], delay [average, min, max, window], ingress points, Egress points, availability, quality degradation refund policy.}

The unitary cost is the cost per month for using the service. The quality of the offered service is measured based on parameters related to throughput and delay. Service availability and average throughput over a window of time are generally the most relevant parameters. Delay is important for real-time services, such as VoIP. Other services may require more stringent parameters such as minimum delay.

A *Quality Degradation refund Policy* (QDP) can be specified at the service level and/or the SLA level. It is a rule which specifies an action to be taken by the service provider in case the offered service quality (availability, throughput, or delay) differs from the one promised.

B. Model of Service, SLA, and Business Indicators

As illustrated in Fig. 3, the overall *business utility* U_b of the set of offered services S to the network operator is modeled as a function of overall monetary profit *P*, services utilities U_s, and market size M_s. *Monetary profit P* is the difference of revenue R_s from contracted services and overall operations cost *C*. M_s is an estimate of future market size based on repurchase intention of current customers, data gathered from competitor providers, as well as input from service usage trends. The utility U_s of an offered service *S* is derived from how satisfied the customers of the service were ($C_s \in [0,1]$), the profit P_s generated from that service, as well as how trendy (T_s) it is compared to others.

$$U_b = w_p P + w_{su} \sum_{s \in S} U_s \times M_s \tag{3.1}$$

$$P = \sum_{s \in S} P_s = \sum_{s \in S} R_s - C \tag{3.2}$$

$$U_s = w_{cp} \times C_s P_s + w_t \times T_s \tag{3.3}$$

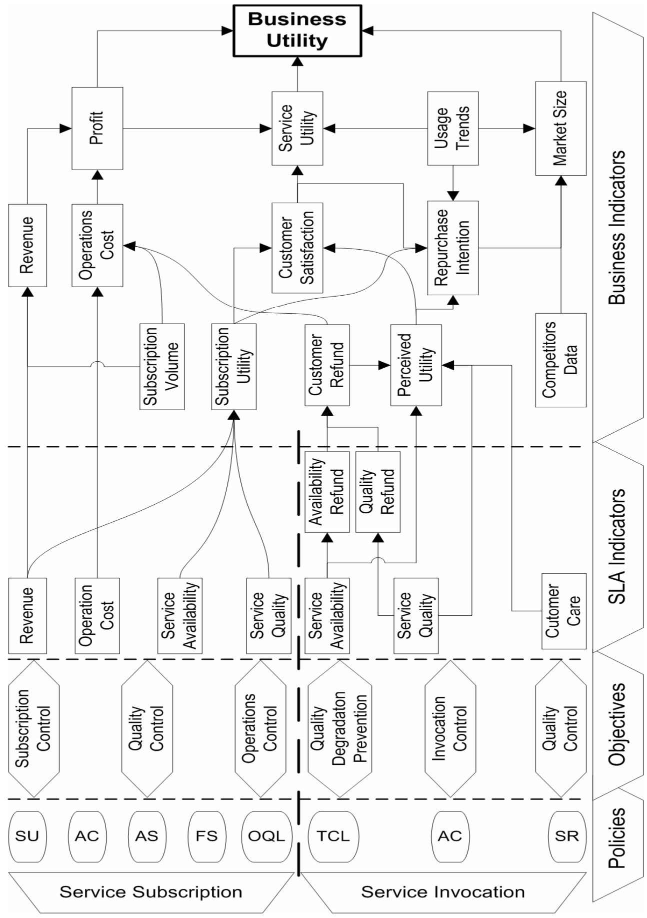


Fig. 3. Model of Service, SLA, and Business Indicators.

Customer Satisfaction C_s for service *s* is derived from the perceived utility U_s^p of *s* and the expected one U_s^e. *Expected utility* U_e is function of contracted service quality, availability av_s, and cost *r* (revenue from the provider's perspective). *Service quality* is directly related to key QoS parameters. We restrict our evaluation of quality to service throughput th_s and delay d_s. The *Perceived Utility* U_p relies on the same parameters in addition to input from customer care C_{cr} quality and refunds of defective sessions C_{rf} if any.

C. Service Management Objectives

The *policy refinement process* relies on a set of high-level objectives specific to the service subscription and invocation management components (Fig.3). In the context of SLS-S, two objectives are considered: (a) controlling the subscription volume (*controlSubscVol*) and (b) controlling the service quality (*controlQlty*). The first relates to policies that set the upper limit in the RAB and those that set the static admission control strategy. Collectively, they regulate the amount of accepted subscription requests. The *controlQlty* objective relates to policies that set service rate factors and OQL and determine the quality of the provided service.

The objectives considered when refining SLS-I policies involve controlling (a) QoS degradation, (b) new invocations, and (c) active services. The first is achieved by policies setting the TCL parameter as this dictates how early proactive actions are taken to prevent quality degradation. Invocation admission control of already subscribed services is achieved by policies setting the AC threshold in the RAB. Finally, SR policies regulate the rates enjoyed by active services.

IV. Analysis of Business-Driven Policies

This section describes the process by which policy parameters are derived from weighted BIs. The derivation process is based on the refinement approach presented in [18].

A. Policy parameters from BIs in the SLS-S component

Profit and *servSatisf* are the BIs influencing SLS-S. Their weights are used in linear mapping functions that derive the relevant policy parameters during the refinement process.

As shown in Fig. 4, the policy setting the SU limit partly achieves the *controlSubscVol* objective, which is influenced by both BIs. Setting the limit close to R_{max} can maximize the subscription volume. This can result in high profit but also in eventual network congestion – due to over-subscription – which can compromise service satisfaction. SU values close to R^{w}_{min} imply minimal risk of congestion but less profit due to lower subscription volumes. It is defined as:

$$SU = R^{w}_{\min} + \frac{1+W_{11}-W_{21}}{2}(R_{\max}-R^{w}_{\min}) \quad (4.1)$$

The *controlSubscVol* objective is also achieved by the policy that sets the TD parameter to be used in the static admission control decision. If TD_{max} is selected (conservative approach) profit suffers, since fewer subscriptions can be accepted, whereas higher satisfaction is achieved because there is confidence that services will be provided at their FS rates. The use of TD_{min} (less conservative approach) however, will result in higher profit, but lower satisfaction since there is confidence that services will only be provided at their AS rates. We define TD as follows:

$$TD = TD_{\max}\text{, when } W_{11} \le W_{21} \quad (4.2a)$$

$$TD = TD_{\min}\text{, when } W_{11} > W_{21} \quad (4.2b)$$

The two indicators also affect policies achieving the *controlQlty* objective. AS factors close to 0 and FS factors close to 0.5 imply increased service satisfaction due to the higher rates produced, but can result in lower profit as fewer service subscriptions can be accommodated. Opposite results are achieved with ASs close to 0.5 and FSs close to 0 because of lower service rates. The final value of individual factors is determined by averaging the result of applying the weights of the BIs, e.g. $Fctr_{AS}= (Fctr_{AS1}+Fctr_{AS2})/2$.

$$Fctr_{AS} = \frac{1}{4}(1+W_{12}-W_{22}) \quad (4.3)$$

$$Fctr_{FS} = \frac{1}{4}(1-W_{12}+W_{22}) \quad (4.4)$$

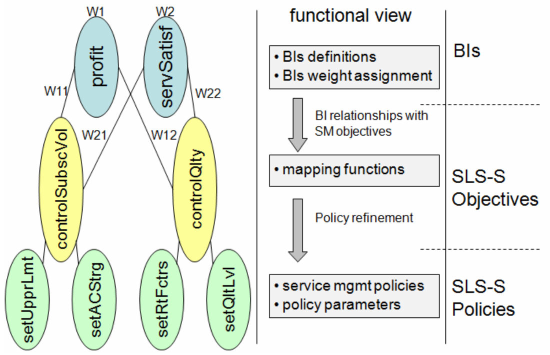


Fig. 4. Relationships between SLS-S BIs, objectives, and policies

The second policy relating to the *controlQlty* objective sets OQL. Values close to 1 suggest that services are provided at their FS rates, even during congestion. This results in higher satisfaction but lower profit since a less number of services can be accommodated within a given RAB. Satisfaction is reduced with OQL values close to 0 as services are most likely to be provided at their AS rates, but an increased profit is achieved with more accepted subscriptions. OQLs close to -1 result in services with no guarantees. Satisfaction is at the lowest possible level whereas profit is at the highest.

$$OQL = (W_{22}-W_{12}) \quad (4.5)$$

B. Policy parameters from BIs in the SLS-I component

Fig. 5 depicts the relationships of the BIs applying to dynamic service management with the associated objectives. As in the case of SLS-S, this section describes the impact of SLS-I policies on the BIs and provides the mapping functions that are used to quantify the policy parameters.

Setting TCL achieves the *control QoS degradation* objective, which is influenced by all three BIs. A TCL close to R_{max} results into delayed QoS degradation prevention actions. This can allow active services to enjoy higher than average service rates for longer and sustain a high probability of accepting new invocations. As such, service satisfaction is maximized, and loss due to invocation rejections (*invRejct*) is minimized. These conditions, however, may eventually cause network congestion and performance degradation

(*perfDegrad*) resulting into potential heavy penalties. A TCL close to R^a_{min} can have the opposite effect on satisfaction and *invRejct* loss since proactive actions are enforced too early. *perfDegrad* loss is also high in this setting because services are likely to receive less than their contracted average rates. R^w_{min} is considered the optimal TCL value for minimizing *perfDegrad* loss. This can result into average levels of service satisfaction and *invRejct* losses. Functions 4.6 to 4.11 take into account the weights of the three BIs and derive the appropriate TCL value. Two functions are provided per BI reflecting the mapping zones between R^a_{min}–R^w_{min} and R^w_{min}–R_{max}. The final TCL value is derived by using the appropriate function based on the weight, and determining the mean of the three resulting TCL instances. Fig. 6a plots the TCL mapping functions.

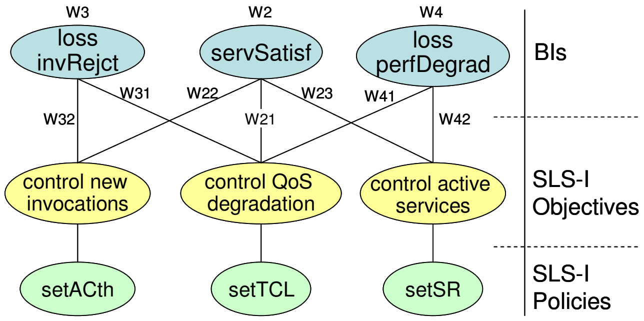


Fig. 5. Relationships between SLS-I BIs, objectives, and policies.

$$TCL_1 = R^a_{\min} + 2W_{21}(R^w_{\min} - R^a_{\min}),\ \text{when } W_{21} \le 0.5 \qquad (4.6)$$

$$TCL_1 = (2R^w_{\min} - R_{\max}) + 2W_{21}(R_{\max} - R^w_{\min}),\ \text{when } W_{21} > 0.5 \qquad (4.7)$$

$$TCL_2 = R^a_{\min} + 2W_{31}(R^w_{\min} - R^a_{\min}),\ \text{when } W_{31} \le 0.5 \qquad (4.8)$$

$$TCL_2 = (2R^w_{\min} - R_{\max}) + 2W_{31}(R_{\max} - R^w_{\min}),\ W_{31} > 0.5 \qquad (4.9)$$

$$TCL_3 = R^a_{\min} + W_{41}(R^w_{\min} - R^a_{\min}),\ \text{when } W_{21} > 0.5 \qquad (4.10)$$

$$TCL_3 = R_{\max} - W_{41}(R_{\max} - R^w_{\min}),\ \text{when } W_{21} \le 0.5 \qquad (4.11)$$

The policy relating to the *control new invocations* objective sets the AC threshold, which is influenced by the *servSatisf* and *invRejct* loss indicators. Values close to R_{max} imply low probability of invocation rejections resulting to minimal losses and increased satisfaction. Low threshold values result to higher losses and less satisfaction due to the increased probability of invocation rejections – R^a_{min} represents the extreme case. Fig. 6b depicts the effect of the BIs weights on the threshold value, which can be determined by taking the mean of AC_{th1} and AC_{th2} from the functions below.

$$AC_{th1} = R^a_{\min} + 2W_{22}(R^w_{\min} - R^a_{\min}),\ \text{when } W_{22} \le 0.5 \qquad (4.12)$$

$$AC_{th1} = (2R^w_{\min} - R_{\max}) + 2W_{22}(R_{\max} - R^w_{\min}),\ \text{otherwise} \qquad (4.13)$$

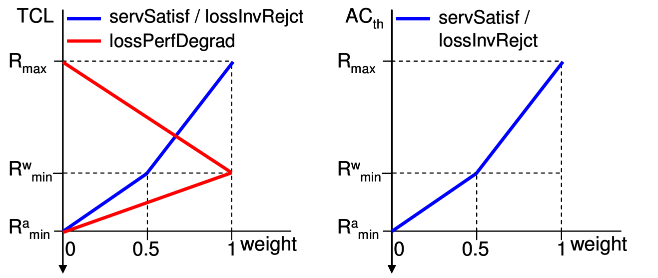
$$AC_{th2} = R^a_{\min} + 2W_{32}(R^w_{\min} - R^a_{\min}),\ \text{when } W_{32} \le 0.5 \qquad (4.14)$$

$$AC_{th2} = (2R^w_{\min} - R_{\max}) + 2W_{32}(R_{\max} - R^w_{\min}),\ \text{for } W_{32} > 0.5 \qquad (4.15)$$

The last SLS-I policy involves service rate adjustments of active services, which have an impact on the user's perceived service quality and on the penalties applying as a result of performance degradation. Values close to SR_{FS} imply high levels of satisfaction and prevention of penalties. AS rates can lead to the reverse due to unfulfilled contracted rates. The impact of the two BIs are quantified by the functions bellow:

$$SR_1 = SR_{AS} + W_{23}(SR_{FS} - SR_{AS}) \qquad (4.16)$$

$$SR_2 = SR_{AS} + W_{42}(SR_{FS} - SR_{AS}) \qquad (4.17)$$



(a) (b)

Fig. 6. Impact of SLS-S BIs weights on (a) TCL, and (b) AC threshold.

By specifying the importance of BIs with weights and using the described mapping functions, a network can be configured according to the business objectives. An ISP may, for example, opt to minimize at the most the penalties for high revenue-generating services, or to build up its reputation through good levels of service satisfaction.

V. Experimental Analysis

All experiments were conducted with a modified OPNET toolkit, extended to support the execution of SLS-I policies. The objective is to analyze the influence of weighted BIs on the enforcement of SLS-I policies in DiffServ/MPLS networks.

A. *Services and Network Topology*

We consider a network service provider that implements the business-driven DiffServ management approach described in the paper, offering SLAs with bundles of FTP, web browsing and email services. The provider classifies services into medium (medQlty) and high quality (highQlty), assigning them to AF11 and AF43 QoS classes, respectively. We consider 7 SLAs of medQlty services with average rate 3,43Mbps each (24Mbps total avg rate) and 7 SLAs of highQlty services with average rate 1,714Mbps each (12Mbps total avg rate). The network topology is shown in Fig. 7. LER1/LER4 are respectively the ingress/egress points for medium quality services and LER2/LER5 for high quality ones. Medium quality services users are located in sites 1 to 7, and sending/receiving traffic to/from site 22. High quality services users are located in sites 11 to 17, and sending/receiving traffic to/from site 23.

The above information is passed to a dimensioning process resulting to the creation of two Traffic Trunks [ingress/egress, QC, RAB_{TT}], namely TT1=[LER1/LER4, AF11, RAB_{TT1}] and TT2=[LER2/LER5, AF43, RAB_{TT2}]. Following our previous definitions, the calculation of the RAB values R^a_{min}, R^w_{min}, R_{max} considers Almost Satisfied ($Fctr_{AS}$) and Fully Satisfied ($Fctr_{FS}$)

factors and the average rates of the offered services. The provider defines $Fctr_{AS}$=0.3, and $Fctr_{FS}$=0.2 for medQlty services (served through TT1), and $Fctr_{AS}$=0.4, and $Fctr_{FS}$=0.3 for highQlty services (served through TT2). The minimum and maximum Traffic Demand is calculated with the following expressions: TD_{min}=(1-$Fctr_{AS}$)(Total avg rate) and TD_{max}=(1+$Fctr_{FS}$)(Total avg rate). This results to TD_{minTT1}=R^a_{minTT1}=16.8Mbps, TD_{minTT2}=R^a_{minTT2}=7.2Mbps, TD_{maxTT1}=28.8Mbps, and TD_{maxTT2} =15.6Mbps.

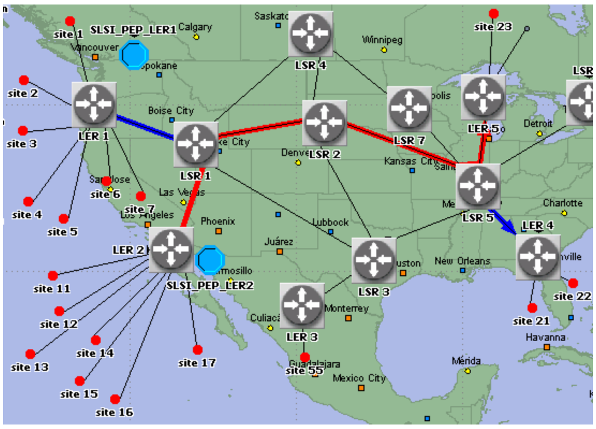


Fig. 7. Network Topology for Experimental Analysis

R^w_{min} is determined after R^a_{min} has been defined for TTs that share physical links. In this topology, TT1 and TT2 share links LSR1-LSR2 and LSR2-LSR5, which are standard E3 links with capacity 34.38 Mbps and represent the maximum resources available along the path for both TTs. To determine R^w_{min} we first determine the available spare resources after the sum of R^a_{min} of the TTs has been subtracted from the capacity of the shared link. In this case the spare resources are 10,37Mbps. These resources are split between TT1 and TT2 proportionally to their TD_{max} value, namely 6.9Mbps and 3.5 Mbps respectively. Consequently R^w_{minTT1} = 6.9Mbps + R^a_{minTT1} = 23.7Mbps. Similarly for TT2, R^w_{minTT2}= 10.6Mbps.

Finally, R_{max} in a TT is calculated by subtracting the sum of R^a_{min} for the TTs sharing resources with the TT, from the capacity of the shared link, namely R_{maxTT1} = 34.38Mbps – 7.2Mbps = 27.1Mbps. Similarly for TT2 R_{maxTT2} = 17.5Mbps.

The values of R^a_{min}, R^w_{min}, R_{max} are significantly important for the enforcement of the business-oriented SLS-I policies. Service rate and admission control adjustments are driven by these values as we show experimentally in the next sections.

B. Business-driven Reaction to Traffic Fluctuations

This section describes the enforcement of business-driven SLS-I policies due to dynamic traffic fluctuations. Consider that invocations of bundled services include the patterns of applications shown in Table III.

Consider a provider that wants to avoid losses due to invocation rejections (*lossInvRejct*) and performance degradation (*lossPerfDegrad*), for which for which he assigns 0.75 to BI weights W3 and W4 (see Fig. 5). The provider can tolerate medium levels of service satisfaction (*servSatisf*), for which a weight of 0.5 to W2 is assigned.

TABLE III
PATTERNS OF APPLICATIONS PER BUNDLED SERVICE

App	Variable	Distribution	Param.	Instances per Bundled Service
http	Page inter-arrival	Poisson	10sec	15/Medium Qty Svc 8/High Qty Svc
	Object Size	Constant	1kB	
	Image	uniform	2kB-10kB	
FTP	Inter-Request time	Poisson	15sec	15/Medium Qty Svc
	File Size	Poisson	500kB	8/High Qty Svc
email	Inter-arrival time	Poisson	15 secs	15/Medium Qty Svc
	E-mail Size	Poisson	4kB	8/High Qty Svc

Policy parameters are derived making use of the mapping functions elaborated in Section IV, and the RAB values described in Section A. For example, expressions 4.6 to 4.11 are used to derive the TCL parameter values. Provided that W2=0.5, TCL_1 is calculated with expression 4.6 (W21≤0.5). For TT1, the computation of TCL_1 results to 23,7Mbps. TCL_2 gives 25.44Mbps and TCL_3 results to 24.5Mbps with expressions 4.9 and 4.11, respectively. Taking the arithmetic mean of the three TCL values, we derive the TCL value of 24.5Mbps for TT1. The derivation of all policy parameters based on the mapping functions of Section IV.B results to the policies summarized in Table IV, which are enforced on routers LER1 and LER2 in Fig. 7.

TABLE IV
SLS-I POLICY ACTIONS

Business Indicators	Policy Actions	
W2=0,5	setTCL(TT1, 24.5Mbps)	setTCL(TT2, 12.3Mbps)
W1=0,75	setSR(TT1, 24.3Mbps)	setSR(TT2, 12.45Mbps)
W2=0,75	setACth(TT1, 24. 5Mbps)	setACth(TT2, 12.3Mbps)

Taking into account the patterns of applications per bundled services (Table III), Fig. 8 shows the throughput of TT1 and TT2, respectively, without the business-oriented policies. The Fig. 8 shows 5 minutes during which the provider serves excessive invocations. The bottom part of the figure shows the amount of invocations received and the active services during each period of the scenario execution. medQlty services are invoked every 20sec and they are active for 70sec. highQlty services are invoked every 40sec and remain active for 80 sec.

Serving excessive invocations at a time (e.g. up to 28 medQlty services, plus up to 14 highQlty services) results into network over-utilization and eventually to congestion. Without control of any kind, service satisfaction is severely reduced for both medQlty and highQlty services. In addition, potential penalties due to performance degradation should be high due to network over-utilisation and congestion during most of this scenario, severely affecting the business value of the provider.

Fig. 9 shows the behaviour of the network when driven by the business-oriented policies introduced earlier. For comparison purposes, in this execution run the patterns of the applications shown in Table III have been used, producing similar patterns of traffic as the ones in the previous experiment (Fig. 8). The number of service invocations and the period in which services are active are also the same as in the previous execution run. The actual service invocations, active services and rejected invocations during the execution of this policy-driven scenario are shown at the bottom of Fig. 9.

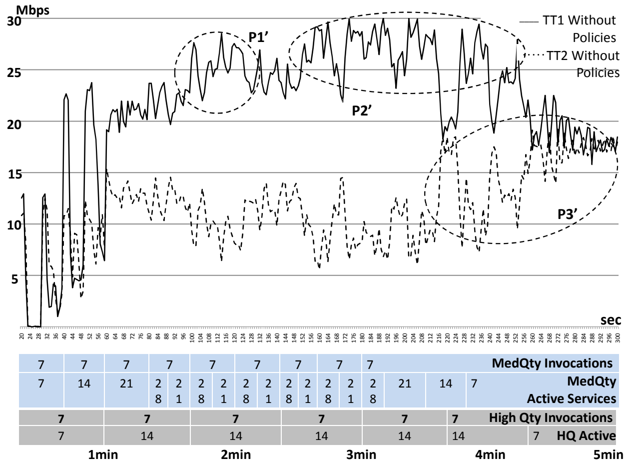


Fig. 8. Scenario execution without the business-oriented policies

The business-driven policies force the network to exploit the resources, but at the same time to protect it for eventual service degradation. For example, up to 28 medQlty services are served before sensible proactive actions are taken as a result of TCL crossings in TT1 (see P1 in Fig. 9). A total of 14 invocation rejections and service rate adjustments result to a sensible reduction of the traffic injection by users, around TCL for TT1 (see P2 in Fig. 9 compared to P2' in Fig. 8). The system has been forced to serve between 14 to 7 medQlty services, which is still highly profitable provided that the dimensioning estimations were originally set to 7 medQlty services for TT1. This way the potential losses due to invocation rejections are reduced. User satisfaction is in general acceptable as users enjoy rates close to TCL, above R^{w}_{minTT1} (23.7Mbps). Potential losses due to performance degradation are also low. In the execution run without policies the resources are over-exploited thus jeopardizing service satisfaction and consequently affecting the business value of the provider (see P1' and P2' in Fig. 8).

Similar results are obtained for high quality services served through TT2. An interesting situation in TT2 is presented as a result of the enforcement of TT1's policies. Due to the reduction of traffic injection in TT1, the DiffServ mechanisms allow traffic served through TT2 to be slightly increased, see for example P4 and P5 in Fig. 9. The same principle is applied in these conditions, proactive actions are taken when traffic goes beyond TCL for TT2, resulting to 14 invocation rejections of high quality services along the scenario execution and the appropriate service rate adjustments, resulting in an eventual normalisation of traffic injection in TT2 (see P3 in Fig. 9). This way the sensible increase of traffic injection exhibited in the normal DiffServ scenario are avoided (P3' in Fig 8). The standard DiffServ mechanisms are concealed with the business oriented approach applied here to increase the business value of the network, serving between 7 to 14 high quality services and taking care of the quality of the services.

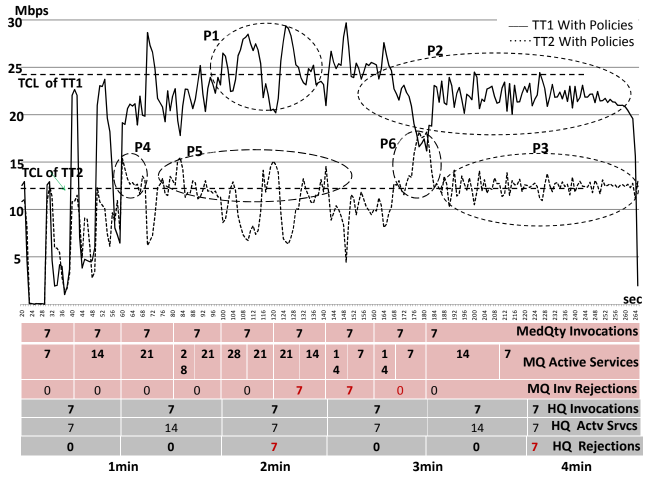


Fig. 9. Scenario execution with the business-oriented policies

In order to show the effectiveness of our approach, traffic fluctuations have been accentuated by disabling the traffic shaping mechanisms of edge routers. Also, to avoid instability due to short traffic fluctuations, we used a 5 second time window for TCL-crossings. This is exemplified in P4 in Fig. 9 where the actual policies are not triggered therein (TCL crossing<5sec).

VI. Related Work

In the context of ISP upgrade and planning operations [22] investigated models linking network service performance, customer behaviour, and market dynamics to profit. Customer relations and profitability were the subject of significant research in market science and economics [16,17,21]. Work in [15] suggests strong relationships among satisfaction, perceived quality, and the disconfirmation with expected quality. [4] follows with a descriptive model relating service quality to customer repurchase intention.

SLA modelling from a policy-based enforcement perspective has been studied in [2]. In [3], a detailed business-driven SLA refinement and optimization into low-level policy configuration is presented, with a case study on incident management in utility data centres. Only raw monetary profit was used as a BI. In this paper we provided a more elaborate policy-based framework for BIs with a link to SLA and service indicators. [19] uses a policy-based dynamic mapping of application QoS parameters to DiffServ classes to provide time and location based services access control and differentiation

so as to maximize utilization and enforce differential service priorities in enterprise nomadism scenarios.

Prior work on incident management, independent of any policy-based solution, was used to assist service personnel to prioritize the processing of action-demanding quality management alerts as per provider's service level management objective [7][5]. Prioritization of service incidents based on their business impact and urgency were proposed. Our work lays down the concepts and provides proof of feasibility of business-driven management into DiffServ QoS domains, identifying the necessary elements for its realization using a rich policy-based business and SLA-driven modelling and refinement. To the best of our knowledge, no prior work addressed business-driven management in DiffServ QoS domain, from the definition of key BIs to their influence on network and service objectives, to the systematic derivation of policies and associated parameter values.

VII. CONCLUDING REMARKS AND FUTURE WORK

This paper described elements required to bridge the gap between business and configuration management in the DiffServ QoS management domain, elaborating on a bidirectional approach that eases the analysis of business-driven DiffServ management strategies.

The approach relies on business indicators that are used by ISPs to derive a network configuration that is in line with high-level business objectives. Such a holistic approach requires the consideration of both static and dynamic admission control of services. In this respect, the core contribution of the paper is the definition of mapping functions and their use by a refinement process to derive the appropriate service management policies, based on the importance of BIs and their relationship with service management objectives.

In this paper we analyzed *service satisfaction*, *profit*, and *loss* BIs. Five mappings were defined based on the influence of the first two indicators on service management objectives (SMOs). The functions control the subscription volume and the quality level of services. Similarly, twelve mapping functions have been defined based on the impact of all BIs on SMOs. They control the invocation logic, performance degradation, and the rates of currently running services. The mapping functions defined are rich enough to quantify all the necessary policy parameters of the refined policies.

The practicality and effectiveness of our approach were demonstrated with an illustrative scenario in which business-related and DiffServ networking issues were put into context.

We developed the instruments to analyse business-driven DiffServ management strategies. Future work involves the study of strategies to maximize the business value and the quality of provided services under different service usage conditions. We will also investigate the classification and validation of service subscription and invocation patterns, and their influence under different services usage patterns. Other areas that will be targeted are the study of pricing strategies and customer frustration reactions to service degradation. These two important aspects are captured in the proposed business model but are not yet explored. We expect them to have a direct impact on service usage, service subscription and invocation for network and service providers. This may possibly require the elaboration of more complex business strategies, and possibly the involvement of realistic models of customer reactions to price and service disruptions.

ACKNOWLEDGMENT

This work is supported in part by the EU IST-EMANICS Network of Excellence (#26854). A special acknowledgment is given to the OPNET's University Program for their support.

REFERENCES

[1] I. Aib , N. Agoulmine, and G. Pujolle; "A Multi-Party Approach to SLA Modeling, Application to WLANs", IEEE CCNC, Jan 2005, USA.

[2] I. Aib et al.,, "Business-Aware Policy-Based Management", IEEE/IFIP BDIM'06, in conjunction with NOMS 2006

[3] I. Aib and R. Boutaba, "On Leveraging Policy-Based Management for Maximizing Business Profit", IEEE TNSM 2007

[4] E. W. Anderson and M. W. Sullivan, "The antecedents and consequences of customer satisfaction of firms," Marketing Science, 12(2), 1993.

[5] C. Bartolini, M. Salle, "Business Driven Prioritization of Service Incidents", 15th IFIP/IEEE DSOM 2004, USA

[6] E. Borcocci, et al., "Admission control algorithm for aggregated pipes service invocation in multi-domain IP environment," Int. Workshop on Next Generation Networking, Coimbra, Portugal, 2006.

[7] M. J. Buco, et al., "Utility computing SLA management based upon business objectives", IBM Systems Journal, Vol 43, No 1, 2004

[8] M. Carlson, et al., "An Architecture for Differentiated Services," RFC2475, 1998.

[9] M. Charalambides, et al., "Policy conflict analysis for DiffServ quality of service management," IEEE TNSM, 2009.

[10] N. Damianou, et al., "The Ponder policy specification language," IEEE Policy Workshop, UK, 2001.

[11] P. Flegkas et al "A Policy-based Quality of Service Management Architecture for IP DiffServ Networks," IEEE Network 16(2), 2002.

[12] M.P. Howarth, et al., "Provisioning for interdomain quality of service: the MESCAL approach" IEEE ComMag, 43(6), June 2005.

[13] E. Mykoniati, et al.,"Admission control for providing QoS in IP DiffServ networks: the TEQUILA approach" IEEE ComMag, 41(1), 2003.

[14] K. Nichols, V. Jacobson and L. Zhang, "A Two Bit Differentiated Services Architecture for the Internet," RFC2638, 1999

[15] R. L. Oliver and W. DeSarbo, "Response determinants in satisfaction judgements," J. Consumer Research, vol. 14, Mar. 1988.

[16] A. Parasuraman, et al., "A conceptual model of service quality and its implications for future research," J. Marketing, vol. 49, pp. 41–50, 1985.

[17] A. Parasuraman et al "SERVQUAL: A multiple-item scale for measuring consumer perceptions of service quality," J. Retailing, 64, 1, 1988.

[18] J. R. Loyola et al "A Methodological Approach towards the Refinement Problem in Policy-based Management Systems" IEEE ComMag, 2006.

[19] B. Tebbani, I. Aib and G. Pujolle, "SLA-based Dynamic Resource Management in Wireless Environments: An enterprise Nomadism Use Case", IJIPT journal, 3(1), pp. 13-21, 2008

[20] P. Trimintzios, et al., "Service-driven traffic engineering for intra-domain quality of service management," IEEE Network Magazine, 17(3), 2003.

[21] Y.Wang et al "An integrated framework for service quality, customer value, satisfaction: Evidence from China's telecommunication industry," Information Systems Frontiers, 6(4), 2004.

[22] Xiao J. and Boutaba R., "Assessing Network Service Profitability: Modeling From Market Science Perspective", IEEE TON, 15(6), 2007.